63-2-3

294954

AIR FORCE INSTITUTE OF TECHNOLOGY

INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY
UNITED STATES AIR FORCE

SCHOOL OF ENGINEERING

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

AF-WP-O-MAY 62 3,000

THESIS

Presented to the Faculty of the School of Engineering of

The Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

DETERMINATION OF THE OPTIMUM DRIVING FUNCTION FOR THE LINEAR CONTROLLER OF A RELAY SERVO LOOP EMPLOYING STATE SPACE TECHNIQUES

By

William Bernard Goggins, Jr. B. S.

Captain USAF

GE/EE/62-8

Graduate Electronics December 1962

AF-WP-O-JAN 63 29

Preface

This report is a summary of my study of State Space optimization techniques. My purpose was to demonstrate the use of these techniques by applying them to a typical adaptive control system.

This study was suggested by Capt. F. M. Brown, an Assistant Professor in the Department of Electrical Engineering at the Air Force Institute of Technology. The study was further prompted by the growing use of State Space techniques in the field of control systems analysis.

I would like to acknowledge the support and assistance given me by Capt. R. A. Hannen, my thesis advisor and Assistant Professor in the Department of Electrical Engineering at the Institute of Technology.

Greatful acknowledgement is also give to Mr. P. B. Brentani of the Minneapolis-Honeywell Regulator Company for the helful explanations and information which he sent me.

Finally, I would like to point out that the mathematical derivations presented in Chapters III and IV assume a knowledge of matrix algebra on the part of the reader.

William B. Goggins, Jr.

Contents

List of Figures

List of Tables

List of Symbols

$\underline{c}$	Initial Condition Vector
$\underline{F}$	Matrix of System Coefficients
$\underline{g}$	Vector of Driving Function Coefficients
h_{ik}	Change in State Variable for a Change in Control
$\underline{H}$	Matrix of h_{ik} s
k	Control Subinterval Index
K	Number of Control Subintervals
n	Order of System
$\underline{q}$	Gradient Vector
$\underline{R}$	Matrix of Terminal Norm Weighting Constants
T	Control Interval
u	Driving Function
V	Terminal Norm
$\underline{x}$	State Vector
γ	Maximum for Absolute Value of Driving Function
δu	Change in Driving Function
δV	Change in Terminal Norm
η	Optimum Distance to Descend
θ_c	Commanded Pitch Rate
θ_m	Model Response
θ_o	System Output
ε	System Error
τ	Control Subinterval
$\underline{\Phi}$	Fundamental Matrix

Abstract

Interest in the field of time optimization of control systems has led to the development of a great number of theories concerning this subject. Many of these have been found difficult to apply to a practical system. However, Ho and Brentani have developed a way to compute the time optimum driving function to be applied to a linear controller by the method of Steepest Descent.

The method of Ho and Brentani was applied to the linear controller of a practical relay servo loop in order to compute the time optimum driving function for this loop. The relay servo loop was also simulated on an analogue computer; the results of simulation were compared with the time optimum driving function. It was found that the two results correlated well.

I. Introduction

One of the most interesting and challenging problems in control system engineering is the application of optimization techniques to an automatic control system. Throughout the years many optimization techniques have been employed and optimizations of several different criteria have been attempted. This report will be concerned with time optimization of a system using State Space techniques. By time optimization is meant that a system is constructed or operated in such a way as to drive any system error to zero in the least time. The system to which these optimization techniques will be applied is an adaptive control system developed by the Minneapolis-Honeywell Regulator Company. This system is set forth in WADC Technical Report 57-349 (Ref 8:85). Since the concepts of adaptive control, of time optimization and of State Space techniques are not easily definable in a sentence or two a further general discussion of the three will be one of the subjects of this chapter. However, a brief statement of the problem is in order at this point.

Statement of the Problem and Purpose

The basic problem considered in this study is the determination of a time optimum driving function to be applied to the linear controller of an aircraft. A comparison of this optimum driving function to that actually used during control of the same aircraft will then be made. From this comparison conclusions will be drawn pertaining to the validity of the optimum solution and to the effectiveness of the

physical system. With this in mind, it might be well to consider, very generally, the concepts of adaptive control, of time optimization, and of State.

The Adaptive Concept

First, a brief discussion of the adaptive concept is in order. In a WADC Technical Report an adaptive control system is defined as one "which has the capability of changing its parameters through an internal process of measurement, evaluation, and adjustment, to adapt to a changing environment, either external or internal, to the vehicle under control (Ref 9:2)."

One way in which such an adaptive system may be realized is shown in Figure I.

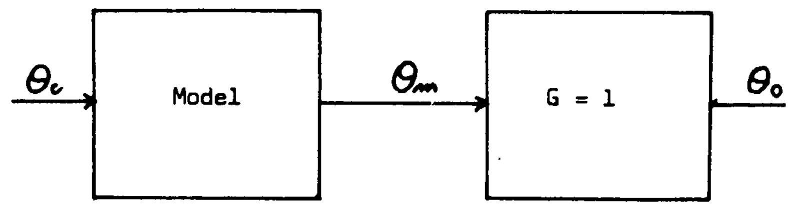

Figure 1
A Hypothetical Adaptive Control System

The command signal θ_c is fed into the model, the response of which is the response desired for the overall system. The rest of the system is then designed to have a transfer function of unity. For example,

an aircraft adaptive autopilot would consist of the following two basic elements: (1) a model, which would have the desired transfer function of the autopilot-aircraft combination, and (2) the actual autopilot-aircraft combination which would be designed to have a transfer function of unity at all times. Thus the output of the entire system is that of the model, and the desired response is obtained. However, since G includes the aircraft, the dynamics of which are constantly changing throughout its flight envelope, other parameters of G must be constantly changed in order to keep G as close to unity as possible. The changing of these parameters through internal measurement, evaluation, and adjustment is the adaptive process.

Time Optimization and the Adaptive System

Obviously it is impossible to physically realize a G of unity. However, if a system could be constructed which would respond in minimum time to the output of a model, a G approximating unity would be realized. A general theory covering this problem was developed in a series of papers by three Russians, Pontriagin, Boltyanskii, and Gramkrelidze, and is known as the maximum principal of Pontriagin. (Ref 11:863) Application of the maximum principal to a linear system shows that if the driving function of the system is "bang-bang", that is, at all times as large, in one direction or the other, as the stops permit, then the system will be time optimum. Such a driving function can be obtained from a relay. A further obvious limitation is that a proper switching sequence for the relay must be provided.

An adaptive system based on the maximum principal might therefore be constructed as in Figure 2.

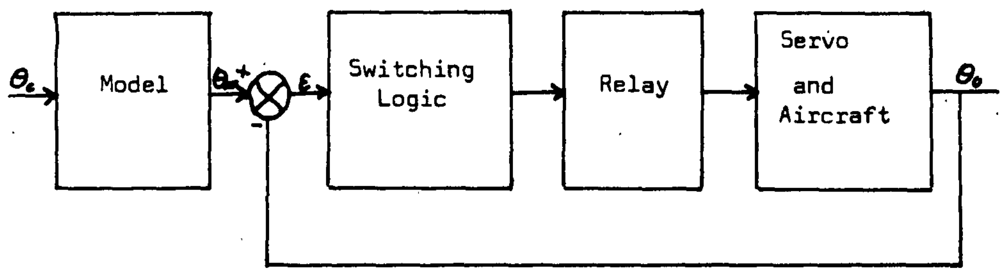

Figure 2
A Practical Adaptive System

In this case the output of the model is fed into a loop which consists of a switching logic, a relay, and a servo and aircraft, all enclosed by unity feedback. The portion of the system of Figure 2 enclosed by the feedback loop will be termed the <u>relay servo loop</u> and will be referred to as such throughout the remainder of this report.

The question of when to switch the relay to obtain the optimum response for the above system must still be answered. To date, a practical optimum switching logic has not been synthesized. However, for the linear portion of the system consisting of the servo and aircraft one can compute relay switching times to drive a given error to zero in minimum time. This is done by a method of successive approximations which will be treated in detail in Chapters III and IV.

The Concept of State

A third consideration is the concept of State, since techniques associated with this concept will be used to determine the optimum switching times for the aforementioned relay (Ref 11:859-860).

Very generally, one might consider that the variables of a system consist of three types of time varying function:

1) a controllable (input) variable $u(t)$ which can be changed at will during the control interval $t > t_0$;

2) an initially controllable (state) variable $x(t)$ which can be chosen at $t = t_0$ but thereafter will act as a function of $u(t)$, the input;

3) an observable (output) variable $y(t)$.

When the first two of these three functions are known the system can be characterized as a function of both of them. Also, the output can be uniquely determined as a function of both. This concept may be expressed in equation form:

$$x(t) = f\left[x(t_0);\ u(t_0, t)\right] \tag{1}$$

$$y(t) = g\left[x(t);\ u(t)\right] \tag{2}$$

In the case of a system of differential equations they assume the form

$$\dot{x}(t) = f\left[x(t), u(t)\right] \tag{3}$$

$$y(t) = g\left[x(t), u(t)\right] \tag{4}$$

A particular case of the above is the matrix equations

$$\dot{\underset{\sim}{x}} = \underset{\sim}{A}\underset{\sim}{x} + \underset{\sim}{B}u \tag{5}$$

$$\underset{\sim}{y} = \underset{\sim}{D}\underset{\sim}{x} \tag{6}$$

where A, B, and D are non-singular constant matrices.

Any linear transfer function which can be expressed in differential equation form can be written in the form of (5) by a suitable transformation. The resulting equation is then the state equation of the system.

A specific example of a tranformation from a transfer function to State Space notation is in order. Consider the transfef function

$$\frac{e_o}{e_{in}} = \frac{1}{s^2 + as + b} \tag{7}$$

The associated differential equation is

$$\frac{d^2 e_o}{dt^2} + a\frac{de_o}{dt} + b e_o = e_{in} \tag{8}$$

Let

$$u = e_{in} \tag{9}$$

$$x_1 = e_o \tag{10}$$

$$x_2 = \dot{x}_1 \tag{11}$$

Substituting (9), (10), and (11) into (8) yields

$$\dot{x}_2 = -b x_1 - a x_2 + e_{in} \qquad (12)$$

In matrix notation the equations are expressed as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -b & -a \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad (13)$$

Several points of interest of the State Space notation should now be explained. If, at any instant x_1 and x_2 are plotted on mutually perpendicular axes, the resultant is a two dimensional vector. This can be extended of course for an N dimensional system into N dimensional Euclidian space; this fact is the origin of the terms State Space and State Vector.

As time is allowed to vary, the State Vector then describes a path in the N dimensional space. Noting that the components of the State Vector are the succesive derivatives of the first component, one sees that for a second order system this plot in a plane is the familiar phase plane which is quite popular for the analysis of non-linear systems. It seems reasonable that the phase plane analysis could somehow be extended to N dimensional phase space.

However, State Space analysis is useful for other reasons. One of the more important of these of the transformation of the system equations into matrix form which facilitates a further transformation from one coordinate system to another, that is from N coordinate space to M coordinate space. The use of matrices also

facilitates programming of a digital computer. In view of the above discussion a detailed statement of the scope of investigation and plan of developement can be made.

Scope of Investigation and Plan of Development

An adaptive control system employing a model followed by a relay servo loop was designed, built and flight tested in an F94C aircraft by the Minneapolis-Honeywell Regulator Company in 1957 as mentioned on page 1 of this report. The system was a pitch axis controller. This report will analyze the system by State Space techniques in order to compute the time optimum driving function for the linear portion of the relay servo loop. A development of these techniques will first be presented, however. The results of an analogue computer simulation of the relay servo loop will then be used as a basis for comparison.

In Chapter II a brief qualitative explanation of the system will be presented in order to explain the operation of the basic system and to set the stage for the analysis of the succeeding chapters. In Chapter III the mathematical theory concerning the solution for the optimum driving function for the linear portion of the relay servo loop will be treated in detail. Chapter IV will then describe the digital computer programs used. Chapter V will present the quantitative analysis of the Honeywell controller by the methods of Chapter III. An analogue computer simulation and its results will be set forth in Chapter VI to form a basis for comparison of the results of the preceding section. Comparison of these results and

conclusions will be left to Chapter VII. Related to Chapters III and IV will be three appendices. Appendix A will explain in detail the various FORTRAN statements of the digital computer programs; in Appendix B comparative solutions of an example problem simple enough to be worked out by hand will be included; Appendix C will show the solution of a typical fourth order problem. The numerical results of these last two appendices will demonstrate to the reader the utility and validity of the methods presented. Tabular data will be presented in Appendix D.

II. Description of the Minneapolis-Honeywell Adaptive System

Before analysis of the Minneapolis-Honeywell Adaptive Control System by State Space techniques it would be well to consider the system in a qualitative manner so that the application of State Space analysis will be clear.

It should be noted here that important symbols used throughout this report are listed and defined in the Table of Symbols in the prefatory part. Figure 3 on the following page is a block diagram of the Honeywell pitch rate system which was installed in the F94C. This block diagram was taken directly from the WADC report (Ref 8:85). The transfer functions of the various components are shown in their respective blocks. These correspond to Flight Condition 13 in the WADC report (Ref 8:153). Flight Condition 13 refers to the F94C aircraft operating at an altitude of 35,000 feet and at a mack number of .73. Note that θ as used in this chapter refers to pitch <u>rate</u>. The operation of the pitch rate system is discussed in the following section.

Operation of the Adaptive System

The basis for operation of the adaptive control system shown in Figure 3 in the forcing of the aircraft pitch rate response θ_a to follow the response of the model θ_m as closely as possible. The model is a simple electrical network with response characteristics equal to those a pilot would desire over the entire flight envelope

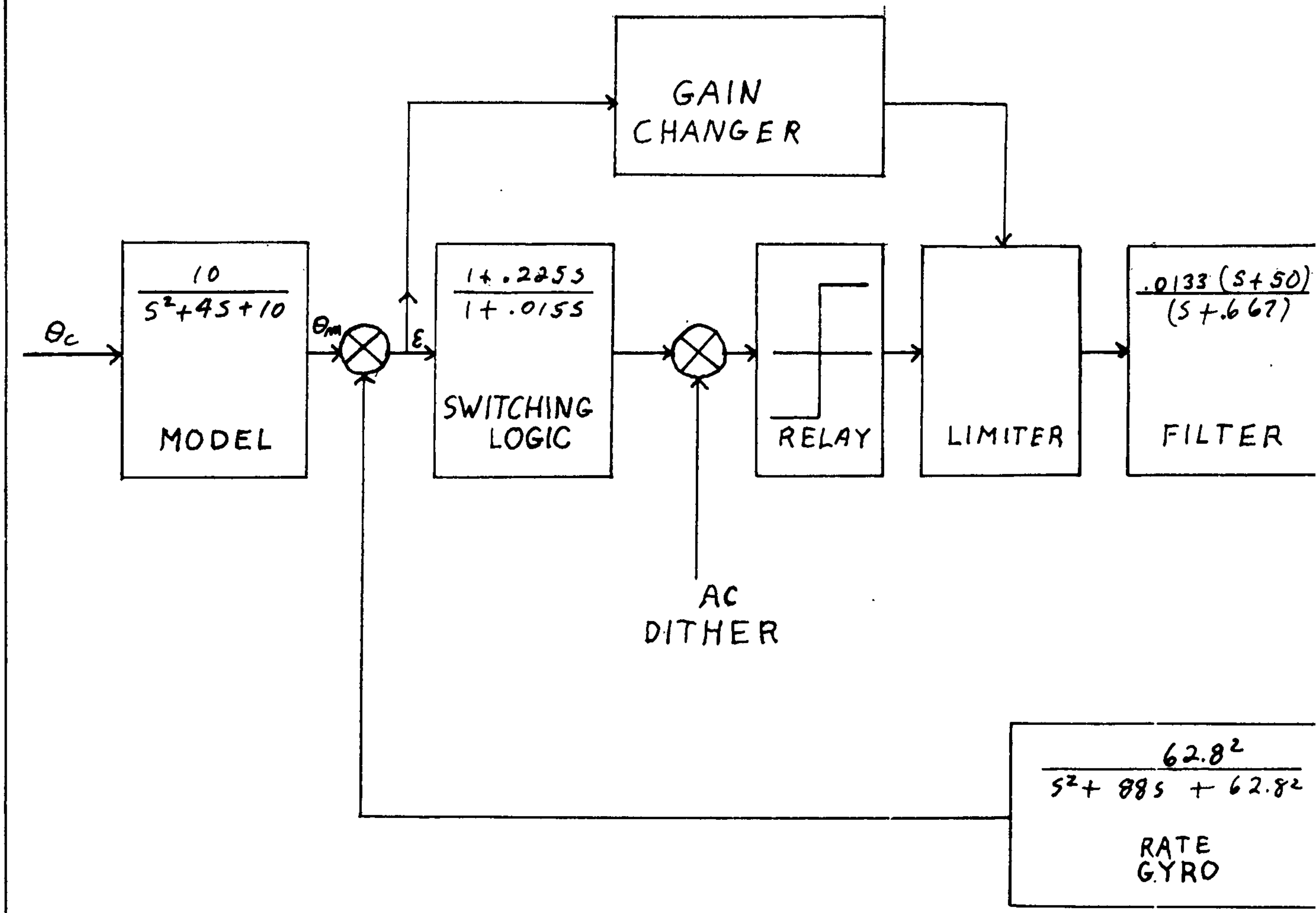

Figure 3
Block Diagram of Minneapolis-Honeywell Adapt
Used in F94-C

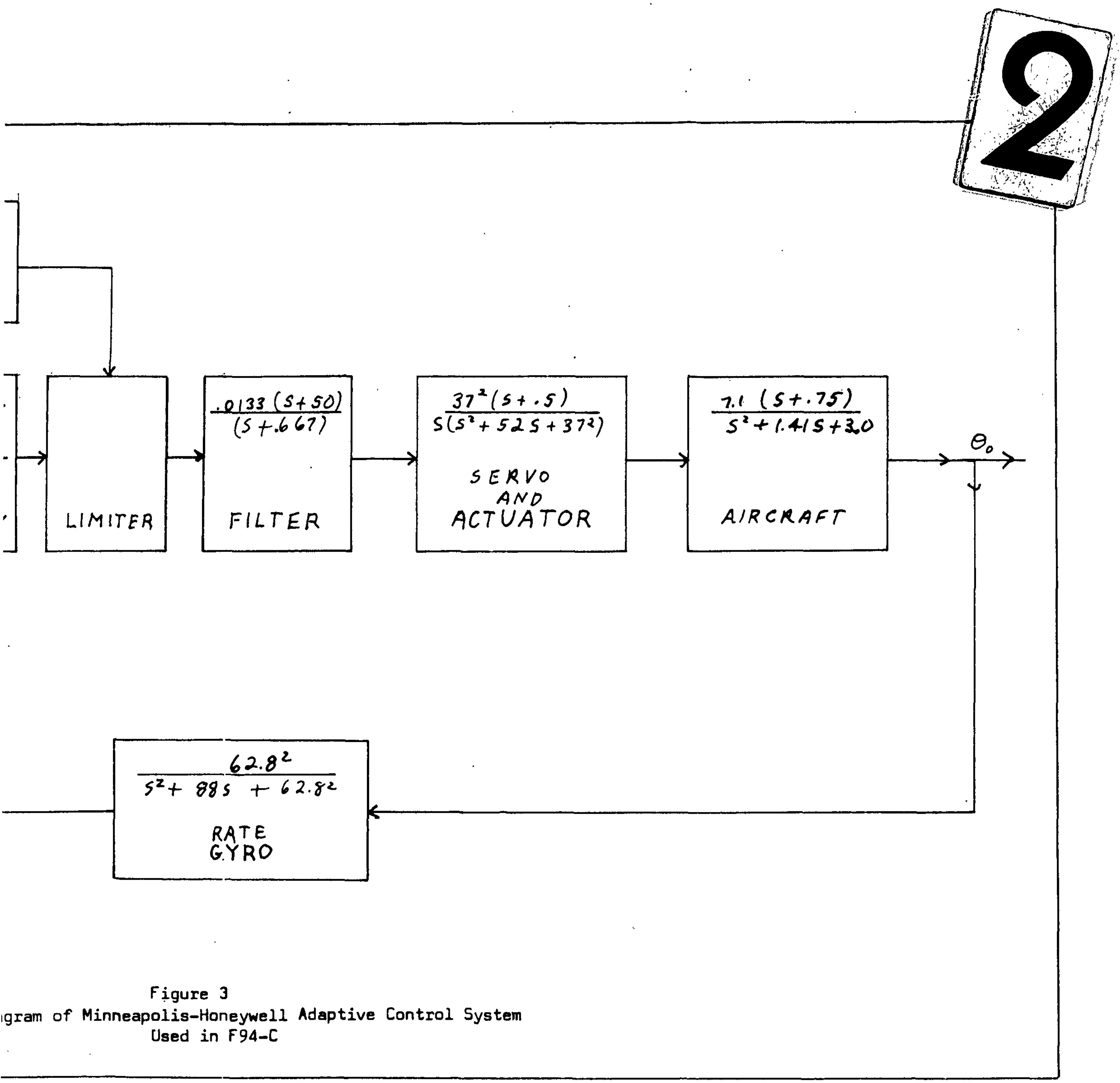

Figure 3
gram of Minneapolis-Honeywell Adaptive Control System
Used in F94-C

of the aircraft. A second order function with a damping ratio of .63 and a natural frequency of 3.17 is used. The desired model response is fed into the relay servo loop. The term <u>relay servo loop</u> here refers to that portion of the system enclosed by the rate gyro feedback, and includes the rate gyro itself. The model output is summed with the negative of the aircraft pitch rate which is fed back by means of the rate gyro. The rate gyro's characteristics are such that this is essentially unity feedback. An error signal $\mathcal{E}$ is thus produced. The error signal is fed through the switching logic, which, as will be shown in Chapter V, is approximately a proportional plus derivative network. The algebraic sum of the error signal and of its derivative multiplied by .225 controls the switching of a very sensitive electronic relay. If the sum is positive the output of the relay will be positive. A 2000 cycle/second sinusoidal dither signal is applied to the relay in order to improve its characteristics. The gain changer and the limiter provide a means for cutting down the relay output voltage for small error signals. In addition the limiter assures proper limiting of the output of the electronic relay.

The filter, the servo and actuator, and the aircraft compose the linear controller portion of the system. Transfer functions shown for the servo and actuator and for the aircraft were derived experimentally. The filter is a lag-lead network used to improve the response of the linear system. The denominator provides a pole close to the zero of the servo and actuator thus preventing this zero from greatly

affecting the response of the loop; the numerator assures that roots due to the servo and actuator poles do not become dominant.

The relay servo loop of the Honeywell adaptive system closely resembles that of the practical adaptive system depicted in Figure 2 in the introductory chapter. The differences are the added components of the Honeywell loop; these are the limiter, the gain changer, and the AC dither. These components provide second order improvements to the system and their inclusion complicates the analysis unnecessarily. Thus they will not be included in any further analysis of the system.

Also, the response of the model can easily be obtained by the Laplace transform technique, or any one of several others. Therefore, only the relay servo loop will be analyzed in this report.

Before going on to Chapter III the reader should note that the filter, the actuator and servo, and the aircraft comprise the linear controller for which an optimum driving function is desired. The next chapter of this report will develop the mathematical theory for finding this optimum driving function.

III. Mathematical Development of the Solution for the Optimum Driving Function

Since the use of State Space techniques and the solution for the optimum driving function are both relatively new this portion of the report will be devoted to a detailed development of the underlying mathematics covering this problem. Because matrix notation will be used freely throughout the development it might be well to define the symbols to be used. A matrix will be represented by an underlined upper case letter, i. e., $\underline{B}$. A vector (n x 1 or 1 x n matrix) will be denoted by an underlined small letter, i. e., $\underline{b}$. A transposed matrix will be denoted by $'$, as $\underline{B}'$. The symbol $^{-1}$ will be used to signify the inverse of a matrix, i. e., $\underline{B}^{-1}$. Successive time derivatives will be denoted by successive dots above the symbol such as $\dot{a}$ or $\ddot{\underline{a}}$. The familiar derivative notation $\frac{d}{dt}$ and the operator D will also be used. The Laplace operator will be denoted by s . Vector or matrix elements will often be written in subscripted form for clarity of explanation. Lower case letters will always be used in this instance.

The first step toward finding the optimum driving function is the development of a transformation which will make it possible to put a general linear transfer function into State Space notation. This transformation will be developed in the next section of this chapter.

Transformation of a General Linear Transfer Function into State Space Form

In the introductory chapter a transformation was introduced to

allow an Nth order differential equation to be written as a series of linear first order differential equations. However, it often occurs in control system problems that the right hand side of the differential equation contains derivatives of the driving function. For instance the transfer function

$$\frac{e_o}{e_{in}} = \frac{s + a}{s^2 + bs + c} \tag{14}$$

becomes

$$\frac{d^2 e_o}{dt^2} + b\frac{de_o}{dt} + ce_o = \frac{de_{in}}{dt} + a e_{in} \tag{15}$$

Since derivatives of many inputs, for instance step or impulse functions, are not calculable in the time domain it is necessary to replace the differential system by a first order system containing no derivatives on the right hand side.

A transformation has been developed by Lanning and Battin (Ref 6:191) for a general time varying system to eliminate right hand side derivatives. Since this paper deals only with the time invariant or constant coefficient case, a formula will be developed inductively considering the coefficients of the differential equation to be constant. Consider the differential equation

$$\frac{d^2 x}{dt^2} + a_1\frac{dx}{dt} + a_2 x = b_0\frac{d^2 u}{dt^2} + b_1\frac{du}{dt} + b_2 u \tag{16}$$

The following system of first order equations is then to be made equivalent to equation (16):

$$x = x_1 + G_0 u \tag{17}$$

$$\frac{dx_1}{dt} = x_2 + G_1 u \tag{18}$$

$$\frac{dx_2}{dt} = -a_2 x_1 - a_1 x_2 + G_2 u \tag{19}$$

Eliminate x_1 and x_2 successively in the following manner:

$$x_1 = x - G_0 u \tag{20}$$

Differentiate (20) with respect to time:

$$\frac{dx_1}{dt} = \frac{dx}{dt} - G_0 \frac{du}{dt} \tag{21}$$

Substituting this result in (18) and rearranging

$$\frac{dx}{dt} = x_2 + G_0 \frac{du}{dt} + G_1 u \tag{22}$$

Now differentiate (22):

$$\frac{d^2x}{dt^2} = \frac{dx_2}{dt} + G_0 \frac{d^2u}{dt^2} + G_1 \frac{du}{dt} \tag{23}$$

After rearranging (23) substitute the result for $\frac{dx_2}{dt}$ in (19). Rearranging the resulting expression then yields

$$\frac{d^2x}{dt^2} - G_0 \frac{d^2u}{dt^2} - G_1 \frac{du}{dt} = -a_2 x_1 - a_1 x_2 + G_2 u \tag{24}$$

Now eliminating x_1 and x_2 by substituting (20) and (22) one has

$$\frac{d^2x}{dt^2} + a_1\frac{dx}{dt} + a_2x = G_0\frac{d^2u}{dt^2} + G_1\frac{du}{dt} + a_1G_0\frac{du}{dt} + G_2u + a_2G_0u + a_1G_1u \tag{25}$$

Equating right hand coefficients of (25) with those of (16) yields

$$b_0 = G_0 \tag{26}$$

$$b_1 = G_1 + a_1G_0 \tag{27}$$

$$b_2 = G_2 + a_2G_0 \tag{28}$$

Solving for G_0, G_1, and G_2 then yields

$$G_0 = b_0 \tag{29}$$

$$G_1 = b_1 - a_1G_0 \tag{30}$$

$$G_2 = b_2 - a_2G_0 - a_1G_1 \tag{31}$$

Inductively, the following relation can then be derived:

$$G_i = b_i - \sum_{k=0}^{i-1} a_{i-k}G_k \; ; \quad i = 1, \cdots, n \tag{32}$$

Thus the N order system

$$\frac{d^nx}{dt^n} + a_1\frac{d^{n-1}x}{dt^{n-1}} + \cdots + a_nx = b_0\frac{d^nu}{dt^n} + \cdots + b_nu \tag{33}$$

may be transformed into a series of first order differential equations of the form

$$x = x_1 + G_0 u \tag{34}$$

$$\dot{x}_1 = x_2 + G_1 u \tag{35}$$

$$\vdots$$

$$\dot{x}_m = -a_m x_1 - a_{m-1} x_2 + \cdots - a_1 x_m + G_m u \tag{36}$$

where G_i $(i = 0, 1, \cdots, m)$ is given by formula (32) and m is the order of the system. In matrix form this is written as

$$\dot{\underline{x}}(t) = \underline{A}\,\underline{x}(t) + \underline{g}\, u(t) \tag{37}$$

$$\dot{\underline{x}}(0) = \underline{c} \tag{38}$$

where $\underline{x}(t)$ is the state vector and $x_1, x_2, \cdots x_m$ are the state variables.

So far a method has been presented for expressing a general linear transfer function in State Space notation. It is now necessary to consider this transformed transfer function in its place as the linear controller of a relay servo loop. As was pointed out in the introduction, it is presently impossible to synthesize a circuit which would perform the switching logic function in an optimum manner. What can be done, however, is the following: given a linear controller as described mathematically by equations (37) and (38) it is possible to compute the driving function $u(t)$ which will reduce the state vector $\underline{x}(t)$ to zero in the shortest possible time. If one lets the state variables represent the error of the system, then one will have computed the

driving function for the linear controller which will reduce the error to zero in the shortest possible time.

Thus, although an optimum switching logic cannot be synthesized the output of such a switching logic can be found. Finding this optimum driving function is a useful tool of analysis since the output of the controller when subjected to this driving function can be found. Comparisons can then be made between these inputs and outputs and those of the same linear controller when utilized in a practical system.

Calculation of the Optimum Driving Function

The problem to be considered in this section is the calculation of the optimum driving function $u(t)$ for the system of equations (37) and (38). A method developed by Ho and Brentani was followed to solve this problem (Ref 5). First, consider a mathematical statement of the problem.

Given a linear system

$$\dot{\underline{x}}(t) = \underline{F}\,\underline{x}(t) + \underline{g}\,u(t) \tag{39}$$

$$\underline{x}(0) = \underline{c} \tag{40}$$

where $\underline{c}$ is an initial error in the system, compute the driving function $u(t)$ necessary to minimize the error in an optimum manner. It will be necessary to state mathematically the criterion or norm used in minimizing the error. However, this statement of the criterion function

will be made in a more appropriate part of this discussion. The general method of solution for the problem must be discussed first.

The Numerical Approximation. Since a closed solution to this problem is in the general case presently unavailable, a numerical approximation will be made and a digital computer will be employed in the computation. Therefore, it will be necessary to divide the control interval into a number of subintervals and make the driving function $u(t)$ piecewise constant. Thus

$$u_k(k\tau) = u(t); \qquad k\tau \leq t < (k+1)\tau; \quad k = 0, 1, \cdots, K-1 \tag{41}$$

$$\tau = \frac{T-0}{K} \tag{42}$$

where τ is the step length and K the number of control subintervals. One must specify the control interval T to be used for each solution. The control interval may also be referred to as the terminal time in this report.

Each u_k now becomes a variable in itself and the problem becomes one of many variables, and of minimizing a function of these variables. To do this, a method of successive approximations known as the Steepest Descent method is employed.

The Steepest Descent Method. Before entering into a mathematical discussion of the method of Steepest Descent an intuitive approach should prove helpful. Figure 4 on the next page is a map of contour lines of a three dimensional surface which is shaped like a bowl. If one were at a point on the rim and wished to get to the lowest

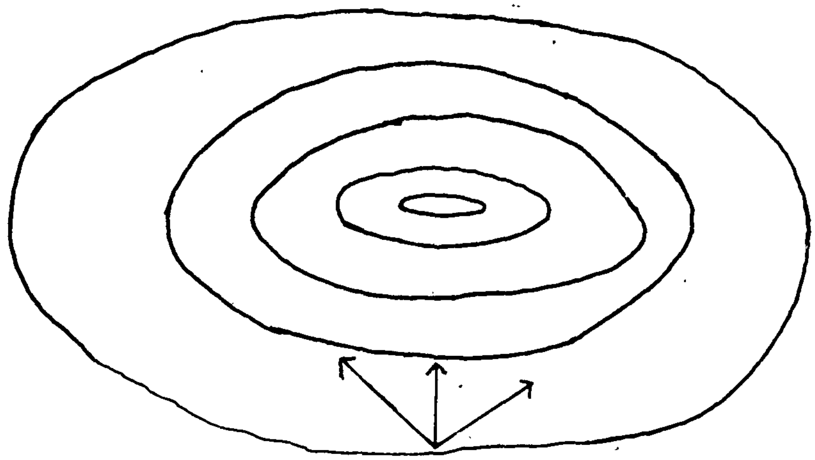

Figure 4
Contour Lines for a Valley

point as fast as possible it would be better to go straight down the slope in the direction of the center arrow than to spiral around the rim in the direction of one of the other two arrows. This is assuming that there aren't too many obstacles blocking the path of descent. It may be deduced even further that even if one cannot see the bottom of the bowl one could continually pick the steepest slope and go down this slope until he gets to the bottom, that is until he starts going up again. The direction taken in this example is the negative of the gradient.

The problem may now be restated mathematically in a more general form. Consider a function to be describable in n dimensional Euclidian

space. One then computes the gradient of the function at some point on a hypersurface in that space. Descending from that point in space along the negative of the gradient then gives the greatest rate of decrease of the function. Although this follows from the previous discussion, a mathematical proof is in order(Ref 7:2-4). Consider a function of several variables $f(x_1 \cdots x_m)$. Start at some point $x_i = x_i(0)$ in m dimensional space and move an infinitesmal distance ds:

$$(ds)^2 = \sum_{i=1}^{m} (dx_i)^2 \tag{43}$$

The total differential of $f(x_1 \cdots x_m)$ is

$$df = \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2 + \cdots + \frac{\partial f}{\partial x_m} dx_m \tag{44}$$

From the above the derivative with respect to s can be expressed as

$$\frac{df}{ds} = \sum_{i=1}^{m} \frac{\partial f}{\partial x_i} \frac{dx_i}{ds} \tag{45}$$

One desires to find the most negative rate of descent $\frac{df}{ds}$ subject to the constraint of (43). In order to do this first rewrite (43) as

$$1 - \sum_{i=1}^{m} \left(\frac{dx_i}{ds} \right)^2 = Z \tag{46}$$

Then adjoin (46) to (45) by means of a Lagrangian multiplier(Ref 4:120-125):

$$V = \sum_{i=1}^{m} \frac{\partial f}{\partial x_i} \frac{dx_i}{ds} + \lambda \left[1 - \sum_{i=1}^{m} \left(\frac{dx_i}{ds} \right)^2 \right] \tag{47}$$

Take partial derivatives in (47) with respect to $\frac{dx_i}{ds}$ and set each equal to zero:

$$\frac{\partial V}{\partial\left(\frac{dx_i}{ds}\right)} = \frac{\partial f}{\partial x_i} - \lambda\left(2\frac{dx_i}{ds}\right) = 0 \qquad i = 1, \cdots, m \tag{48}$$

The following is obtained by rearranging (48):

$$\frac{dx_i}{ds} = \frac{1}{2\lambda}\frac{\partial f}{\partial x_i} \qquad i = 1, \cdots, m \tag{49}$$

Now form the sum of the squares of each of the equations of (49):

$$\sum_{i=1}^{m}\left(\frac{dx_i}{ds}\right)^2 = \frac{1}{4\lambda^2}\sum_{i=1}^{m}\left(\frac{\partial f}{\partial x_i}\right)^2 \tag{50}$$

The left hand side of (50) is seen to be unity from equation (43):

$$1 = \frac{1}{4\lambda^2}\sum_{i=1}^{m}\left(\frac{\partial f}{\partial x_i}\right)^2 \tag{51}$$

Solving (51) for λ yields

$$\lambda = \pm\frac{1}{2}\sqrt{\sum_{i=1}^{m}\left(\frac{\partial f}{\partial x_i}\right)^2} \tag{52}$$

This value for λ can be substituted into (49) to obtain

$$\frac{dx_i}{ds} = \pm\frac{\frac{\partial f}{\partial x_i}}{\left[\sum_{i=1}^{m}\left(\frac{\partial f}{\partial x_i}\right)^2\right]^{\frac{1}{2}}} \tag{53}$$

Recalling that

$$\frac{df}{ds} = \sum_{i=1}^{m}\frac{\partial f}{\partial x_i}\frac{dx_i}{ds} \tag{45}$$

Substituting (45) into (53) after solving for $\frac{dx_i}{ds}$ yields the most negative $\frac{df}{ds}$:

$$\frac{df}{ds} = \pm \sum_{i=1}^{m} \left(\frac{dx_i}{ds}\right)^2 \left[\sum_{i=1}^{m} \left(\frac{\partial f}{\partial x_i}\right)^2\right]^{\frac{1}{2}} \tag{54}$$

The inner summation is performed before the outer summation and is therefore factorable:

$$\frac{df}{ds} = \pm \left[\sum_{i=1}^{m} \left(\frac{\partial f}{\partial x_i}\right)^2\right]^{\frac{1}{2}} \sum_{i=1}^{m} \left(\frac{dx_i}{ds}\right)^2 \tag{55}$$

Again recalling that

$$\sum_{i=1}^{m} \left(\frac{dx_i}{ds}\right)^2 = 1 \tag{43}$$

The second summation becomes 1 and thus

$$\frac{df}{ds} = \pm \left[\sum_{i=1}^{m} \left(\frac{\partial f}{\partial x_i}\right)^2\right]^{\frac{1}{2}} \tag{56}$$

Examination of (56) reveals that these are respectively the positive and negative gradients. The maximum rate of descent, which is the maximum rate of change of the function with respect to distance, is therefore obtained by making the direction of descent equal to the negative of the gradient.

Complete Statement of the Control System Problem. With the above result in mind it is now possible to consider the problem of obtaining a solution for the optimum driving function of the linear controller

by the Steepest Descent method. In an actual autopilot system, an additional constraint must be added to those already stipulated in the preceeding sections. This additional constraint limits the maximum excursion of the driving function to one that will not overdrive the system. A more complete mathematical statement of the problem may now be made. Consider the dynamic system

$$\underline{\dot{x}}(t) = \underline{F}\,\underline{x}(t) + \underline{g}\,u(t) \tag{57}$$

$$\underline{x}(0) = \underline{c} \tag{58}$$

which can be written in subscripted form as

$$\dot{x}_i(t) = \sum_{j=1}^{m} f_{ij}\, x_j(t) + g_i\, u(t) \tag{59}$$

$$x_i(0) = c_i \qquad i = 1, \cdots, m \tag{60}$$

It is now necessary to state the function to be minimized:

$$V = \| x(T) \|^2_{\underline{R}} = \sum_{i=1}^{m} \sum_{j=1}^{m} x_i(T)\, x_j(T)\, r_{ij} = \underline{x}'(T)\, \underline{R}\, \underline{x}(T) \tag{61}$$

where $\underline{R}$ is a positive definite symmetrical matrix of weighting functions. The function (61) will be referred to as the performance function or as the terminal norm. Since V is quadratic in form, finding the minimum V will also find the minimum of $\underline{x}(T)$. The choice of the matrix $\underline{R}$ will be discussed in detail in Chapter IV.

The system will be subject to the maximum input constraint mentioned above:

$$|u(t)| \leq \gamma \tag{62}$$

or for u piecewise constant

$$|u_k| \leq \gamma \qquad k = 0, 1, \cdots, K-1 \tag{63}$$

Determining the Gradient. Now that a complete mathematical statement of the problem has been made the method of solution may be considered. Briefly stated, the method of solution is to compute the gradient of V with respect to u and then to descend along this path subject to the constraint that u may not go beyond a boundary formed by equation (63).

As before define

$$u_k(k\tau) = u(t) \qquad k\tau \leq t < (k+1)\tau \tag{41}$$

$$\tau = \frac{T-0}{K} \qquad k = 0, \cdots, K-1 \tag{42}$$

Thus

$$u = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{K-1} \end{bmatrix} \tag{64}$$

It is necessary to find the change in the terminal norm V with respect to a change in u_k. Recalling that each u_k is a seperate variable in K dimensional space the following relationship can be formed:

$$\frac{\partial V}{\partial u_k} = \sum_{i=1}^{m} \frac{\partial V}{\partial x_i(T)} \frac{\partial x_i(T)}{\partial u_k} \qquad k = 0, \cdots, K-1 \tag{65}$$

This says that for any u_k the change in V with respect to that u_k

equals the change in V with respect to the change in the terminal state vector multiplied by the change in the terminal state vector with respect to the change in the control $\underline{u}$. At this point define

$$\frac{\partial x_i(T)}{\partial u_k} = h_{ik} \tag{66}$$

as an element of an $m \times K$ matrix $\underline{H}$. Making this substitution in (65) yields

$$\frac{\partial V}{\partial u_k} = \sum_{i=1}^{m} \frac{\partial V}{\partial x_i(T)} h_{ik} \qquad k = 0, \cdots, K-1 \tag{67}$$

Performing the indicated summation

$$\frac{\partial V}{\partial u_k} = \frac{\partial V}{\partial x_1(T)} h_{1,k} + \frac{\partial V}{\partial x_2} h_{2,k} + \cdots + \frac{\partial V}{\partial x_m(T)} h_{m,k} \qquad k = 0, \cdots, K-1 \tag{68}$$

In matrix form this is written as

$$\frac{\partial V}{\partial \underline{u}} = \underline{H}' \frac{\partial V}{\partial \underline{x}(T)} \tag{69}$$

Recalling the expression for the terminal norm

$$V = \underline{x}'(T)\, \underline{R}\, \underline{x}(T) \tag{61}$$

For $\underline{R}$ symmetrical the derivative is calculated to be (Ref 3:48)

$$\frac{\partial V}{\partial \underline{x}(t)} = 2 \underline{R}\, \underline{x}(t) \tag{70}$$

Substituting (70) into (69) yields

$$\frac{\partial V}{\partial \underline{u}} = 2 \underline{H}' \underline{R}\, \underline{x}(T) \tag{71}$$

$\underline{x}(T)$ can be calculated by solving the system equations. Thus it is only necessary to determine $\underline{H}$ in order to find $\frac{\partial k}{\partial \underline{u}}$.

Calculation of $\underline{H}$. An examination of the solution to the system equations will lead to a method for calculating $\underline{H}$. The solution to equations (57) and (58) is (Ref 1:78)

$$\underline{x}(T) = \underline{\Phi}(T,0)\,\underline{c} + \int_0^T \underline{\Phi}(T,\sigma)\,\underline{g}\,u(\sigma)\,d\sigma \tag{72}$$

where

$$\frac{d\underline{\Phi}}{dt} = \underline{F}\,\underline{\Phi}(t) \quad ; \quad \underline{\Phi}(0) = \underline{I} \tag{73}$$

where $\underline{I}$ is the identity matrix. To find $\underline{\Phi}(T,\sigma)$ which is under the integral sign the adjoint system may be utilized. This is defined as

$$\frac{d\,\underline{\Phi}'(T,\sigma)}{d\sigma} = -\underline{F}'\,\underline{\Phi}'(T,\sigma) \quad ; \quad \underline{\Phi}(T) = \underline{I} \tag{74}$$

It is now necessary to consider the solutions to equation (73) and to the adjoint equation (74). Equation (73) resembles the scalar form

$$\frac{dy}{dt} = a y \; ; \quad y(0) = 1 \tag{75}$$

The solution of the above scalar equation is known from differential equation theory to be

$$y = e^{at} \tag{76}$$

where the exponential is defined

$$e^{at} = \sum_{n=0}^{\infty} \frac{a^n t^n}{n!} \tag{77}$$

In like manner the solution to equation (73) is

$$\underline{\Phi}(t) = e^{\underline{E}t} \quad ; \quad \underline{\Phi}(0) = \underline{I} \tag{78}$$

if $e^{\underline{E}t}$ is defined as follows

$$e^{\underline{E}t} = \sum_{n=0}^{\infty} \frac{\underline{E}^n t^n}{n!} \quad ; \quad \underline{E}^0 = \underline{I} \tag{79}$$

That this series converges and is differentiable term by term is readily shown (Ref 3:42-45). Term by term differentiation and substitution into equation (73) shows that equation (79) is indeed a solution. The matrix $\underline{\Phi}$ is often known as a fundamental matrix for the system. Similarly the solution to (74) is found to be

$$\underline{\Phi}(T-\sigma) = e^{\underline{E}(T-\sigma)} \tag{80}$$

Substitution of these results into (72) yields

$$\underline{x}(T) = e^{\underline{E}T}\underline{c} + \int_0^T e^{(T-\sigma)\underline{E}}\,\underline{g}\,u(\sigma)\,d\sigma \tag{81}$$

Equation (81) is the same as the solution (72) except that it is written in exponential form. The definitions of the exponentials may be used to calculate a numerical solution for $\underline{x}(T)$. However, this problem will be taken up in Chapter IV. Use of the exponential form will also clarify the following steps in the development of a method for the calculation of the matrix $\underline{H}$.

Suppose for a moment that $u(t)$ were constant over the entire control interval. It then could be factored from under the integral thus facilitating

differentiation with respect to u.

$$\underline{x}(T) = e^{\underline{E}T}\underline{c} + u\int_0^T e^{(T-\sigma)\underline{E}}\underline{g}\,d\sigma \tag{82}$$

The derivative for u constant now becomes

$$\frac{\partial \underline{x}(T)}{\partial u} = \int_0^T e^{(T-\sigma)\underline{E}}\underline{g}\,d\sigma \tag{83}$$

The value of making u piecewise constant now becomes evident. The integration can be broken up, with the interval of each part chosen to be the control subinterval τ; and each u_k can be factored from the integral. If the exponentials are chosen judiciously the following result is obtained:

$$\begin{aligned}\underline{x}(T) = e^{\underline{E}T}\underline{c} &+ u_0\int_0^{\tau} e^{(T-\tau)\underline{E}}e^{(\tau-\sigma)\underline{E}}\underline{g}\,d\sigma \\ &+ u_1\int_{\tau}^{2\tau} e^{(T-2\tau)\underline{E}}e^{(2\tau-\sigma)\underline{E}}\underline{g}\,d\sigma \\ &\;\vdots \\ &+ u_k\int_{k\tau}^{(k+1)\tau} e^{[T-(k+1)\tau]\underline{E}}e^{[(k+1)\tau-\sigma]\underline{E}}\underline{g}\,d\sigma\end{aligned} \tag{84}$$

Now differentiate $\underline{x}(T)$ with respect to each u_k:

$$\frac{\partial[\underline{x}(T)]}{\partial u_k} = \int_{k\tau}^{(k+1)\tau} e^{[T-(k+1)\tau]\underline{E}}e^{[(k+1)\tau-\sigma]\underline{E}}\underline{g}\,d\sigma \tag{85}$$

The exponential $e^{[T-(k+1)\tau]\underline{E}}$ is a constant and may be brought from under the integral:

$$\frac{\partial[\underline{x}(T)]}{\partial u_k} = e^{[T-(k+1)\tau]\underline{E}}\int_{k\tau}^{(k+1)\tau} e^{[(k+1)\tau-\sigma]\underline{E}}\underline{g}\,d\sigma \tag{86}$$

The integral of (86) can be simplified to a much more maneagable form. To do this it is first necessary to evaluate the integral.

To do this perform the integration:

$$\int_{k\tau}^{(k+1)\tau} e^{[(k+1)\tau-\sigma]\underline{F}}\,\underline{g}\,d\sigma = -\underline{F}^{-1}e^{(k+1)\tau\underline{F}}\left[e^{-\sigma\underline{F}}\underline{g}\right]_{k\tau}^{(k+1)\tau} \tag{87}$$
$$\underline{F}^{-1}\left(e^{\tau\underline{F}}-\underline{I}\right)\underline{g}$$

By equating this result with that obtained from evaluation of the following integral a convenient way of calculating the integral of equation (87) can be determined:

$$\int_0^{\tau} \underline{\Phi}(\sigma)\,\underline{g}\,d\sigma = \int_0^{\tau} e^{\sigma\underline{F}}\,\underline{g}\,d\sigma \tag{88}$$

Integration yields

$$\underline{F}^{-1}\left[e^{\sigma\underline{F}}\right]_0^{\tau}\underline{g} = \underline{F}^{-1}\left[e^{\tau\underline{F}}-\underline{I}\right] \tag{89}$$

Equating the results of the two integrations one obtains

$$\int_{k\tau}^{(k+1)\tau} e^{[(k+1)\tau-\sigma]\underline{F}}\,\underline{g}\,d\sigma = \int_0^{\tau} e^{\sigma\underline{F}}\,\underline{g}\,d\sigma \tag{90}$$

Substituting the results of (90) back into equation (86) then yields

$$\frac{\partial[\underline{x}(T)]}{\partial u_k} = e^{[T-(k+1)\tau]\underline{F}}\int_0^{\tau} e^{\sigma\underline{F}}\,\underline{g}\,d\sigma \qquad k=0,\cdots,K-1 \tag{91}$$

Reverting back to the notation of the fundamental matrix one has the following which is equivalent to (91):

$$\frac{\partial[\underline{x}(T)]}{\partial u_k} = \underline{\Phi}(T-(k+1)\tau)\int_0^{\tau}\underline{\Phi}(\sigma)\,\underline{g}\,d\sigma \tag{92}$$

$k=0,\cdots,K-1.$

In order to simplify the notation let

$$\underline{a} = \int_0^{\tau} \underline{\Phi}(\sigma)\, \underline{q}\, d\sigma \tag{93}$$

Recalling that

$$h_{ik} = \frac{\partial [x_i(T)]}{\partial u_k} \tag{66}$$

Putting (92) into matrix notation one has the result

$$\underline{H} = \left[\underline{\Phi}(T-\tau)\underline{a} \quad \underline{\Phi}(T-2\tau)\underline{a} \quad \cdots \quad \underline{a} \right] \tag{94}$$

<u>Distance to Descend</u>. Having determined a means for calculating the gradient it is now necessary to determine the proper distance to descend along this gradient in order to maximize the change in the terminal norm V. Putting the solution to the system equations in a more maneagable form facilitates the algebra involved in finding this proper distance to descend. Recall the solution to the system equations:

$$\underline{x}(T) = e^{\underline{F}T}\underline{c} + \int_0^T e^{(T-\sigma)\underline{F}}\, \underline{q}\, u(\sigma)\, d\sigma \tag{81}$$

Let

$$\underline{P} = e^{\underline{F}T}\underline{c} \tag{95}$$

Then using the result for $\underline{H}$, which is equation (94) and the matrix version of the driving function $\underline{u}$, which is equation (64), a matrix

formula for $\underline{x}(T)$ can be obtained:

$$\underline{x}(T) = \underline{P} + \underline{H}\,\underline{u} \tag{96}$$

Now to continue with the calculation of the proper distance to descend along the gradient consider the terminal norm formula, which is equation (61), with equation (95) substituted:

$$V = (\underline{P}' + \underline{u}'\underline{H}')\,\underline{R}\,(\underline{P} + \underline{H}\,\underline{u}) \tag{97}$$

However, the change in the terminal norm V with a change in the driving function $\underline{u}$ is of interest. Taking $\underline{u}$ at any point, say i, and performing the matrix operations indicated in (97)

$$V_i = \underline{P}'\underline{R}\,\underline{P} + \underline{u}_i'\underline{H}'\underline{R}\,\underline{H}\,\underline{u}_i + 2\,\underline{P}'\underline{R}\,\underline{H}\,\underline{u}_i \tag{98}$$

Consider a small variation in $\underline{u}$:

$$\underline{u}_{i+1} = \underline{u}_i + \delta\underline{u} \tag{99}$$

Substituting (99) into (98) yields

$$V_{i+1} = \underline{P}'\underline{R}\,\underline{P} + (\underline{u}_i' + \delta\underline{u}')\,\underline{H}'\underline{R}\,\underline{H}\,(\underline{u}_i + \delta\underline{u}) + 2\,\underline{P}'\underline{R}\,\underline{H}\,(\underline{u}_i + \delta\underline{u}) \tag{100}$$

Then carry out the multiplication called for in (100)

$$V_{i+1} = \underline{P}'\underline{R}\,\underline{P} + \underline{u}_i'\,\underline{H}'\underline{R}\,\underline{H}\,\underline{u}_i + 2\,\underline{P}'\underline{R}\,\underline{H}\,\underline{u}_i \tag{101}$$
$$+ \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} + 2\,\underline{P}'\underline{R}\,\underline{H}\,\delta\underline{u} + 2\,\underline{u}_i'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u}$$

Now define

$$\delta V = V_{i+1} - V_i \tag{102}$$

Then subtracting (98) from (101) leaves

$$\delta V = 2\,\underline{u}_i'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} + 2\,\underline{p}'\underline{R}\,\underline{H}\,\delta\underline{u} + \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} \tag{103}$$

Combining the first two terms

$$\delta V = 2(\underline{p}' + \underline{u}_i'\underline{H}')\underline{R}\,\underline{H}\,\delta\underline{u} + \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} \tag{104}$$

Substituting equation (96) into (104)

$$\delta V = 2\,\underline{\chi}'(T)\,\underline{R}\,\underline{H}\,\delta\underline{u} + \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} \tag{105}$$

Recalling equation (71) and taking its transpose one has

$$2\,\underline{\chi}'(T)\,\underline{R}\,\underline{H} = 2\left[\underline{H}'\underline{R}\,\underline{\chi}(T)\right]' = \left(\frac{\partial V}{\partial \underline{u}}\right)' \tag{106}$$

Substituting the above result into (105)

$$\delta V = \left(\frac{\partial V}{\partial \underline{u}}\right)'\delta\underline{u} + \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} \tag{107}$$

Now define

$$\frac{\partial V}{\partial \underline{u}} = \underline{g} \tag{108}$$

Finally

$$\delta V = \underline{g}'\delta\underline{u} + \delta\underline{u}'\underline{H}'\underline{R}\,\underline{H}\,\delta\underline{u} \tag{109}$$

Equation (109) is an expression for the variation in the terminal norm verses a small variation in the driving function $\delta \underline{u}$. It is now desirable to maximize the change in the terminal norm δV by determining a proper distance to descend along the gradient $\underline{g}$. The change in the driving function $\delta \underline{u}$ is equal to the direction of descent multiplied by the proper distance to descend:

$$\delta \underline{u} = \eta \underline{g} \tag{110}$$

where η is a constant and is the proper distance to descend.

Were it possible to descend along the gradient without being subject to the constraint on the driving function discussed on page 25, η could be determined in the following manner. Substitute (110) into (109):

$$\delta V = \eta \underline{g}' \underline{g} + \eta^2 \underline{g}' \underline{H}' \underline{R} \underline{H} \underline{g} \tag{111}$$

Calculus can now be employed to determine the value for η which will maximize δV. Differentiate equation (111) with respect to η and set the resulting expression equal to zero:

$$\frac{d(\delta V)}{d\eta} = \underline{g}' \underline{g} + 2\eta \underline{g}' \underline{H}' \underline{R} \underline{H} \underline{g} = 0 \tag{112}$$

Solving for η

$$\eta = \frac{-\underline{g} \underline{g}'}{2 \underline{g}' \underline{H}' \underline{R} \underline{H} \underline{g}} \tag{113}$$

However, δu is limited since $u + \delta u$ must not go outside the boundary which was determined by the maximum contraint on the driving function u. Proper use of this constraint will aid in computing the proper change in the driving function subject to the constraint.

The Descent Schemes. Two different descent schemes may be determined utilizing the constraint on the driving function. These are the Gradient Projection Descent Scheme and the Corner Aiming Descent Scheme. Both schemes use the constraint on the driving function to limit δu but they do it in a slightly different way. A geometrical interpretation of the Gradient Projection Descent Schme is shown in Figure 4a which is a two dimensional model used to aid in the explanation of the descent schemes. Figure 4a is located on the next page. Since this is a two dimensional model only two control subintervals are used. As before, the attempt is made to minimize the terminal norm V by changing the driving function u. The negative of the gradient is therefore drawn from point A across the lines of constant V which are the same as the contour lines of Figure 4. The vector AB thus represents ηg with η computed as in equation (113). However, the boundary for the driving function does not allow a vector of this length. The tip of the vector is therefore projected down to the boundary to point C. A vector drawn from point A to the intersection of the projection, C, is then used as the change in control. It is from this procedure that the term "gradient projection" is derived.

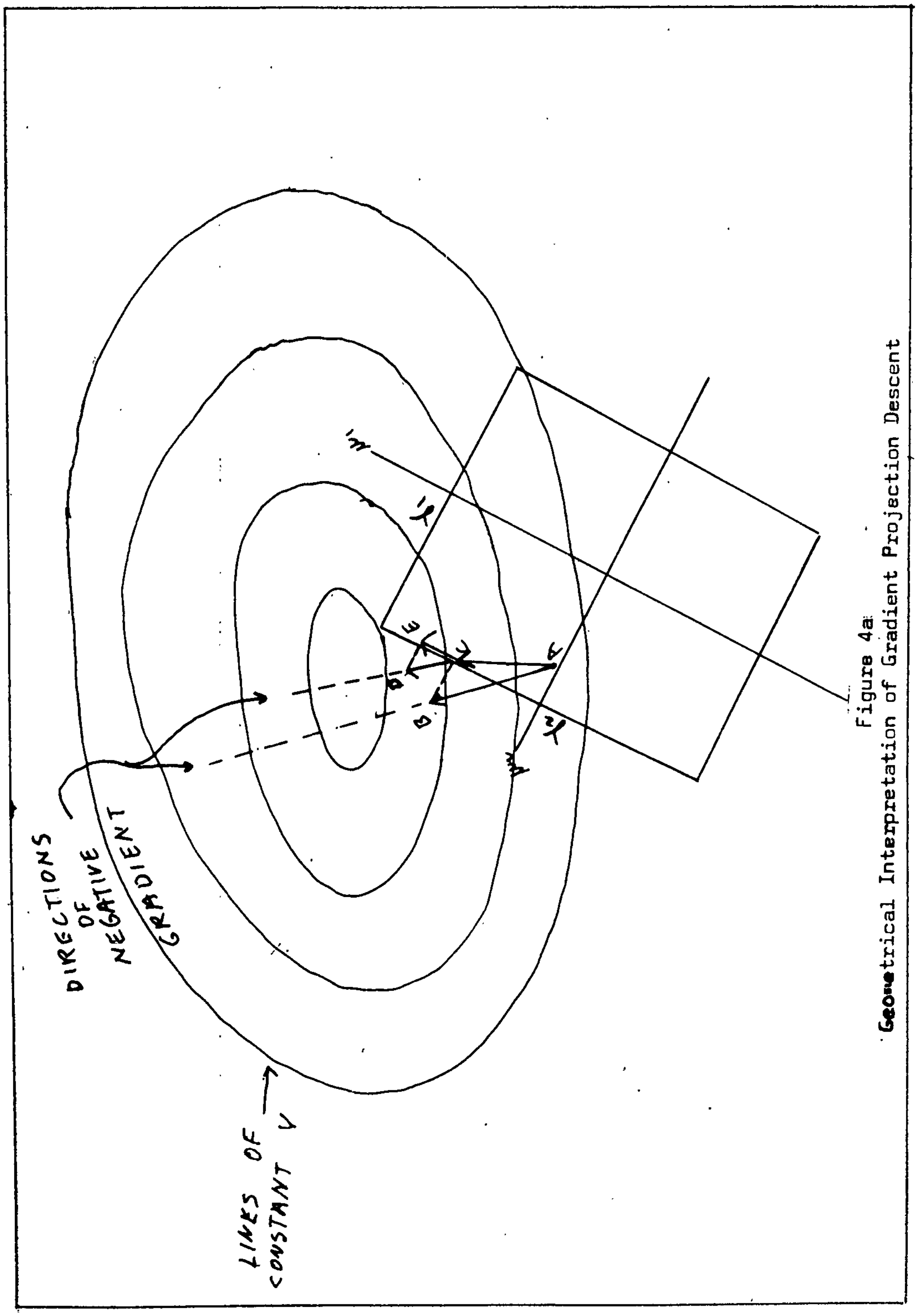

Figure 4a
Geometrical Interpretation of Gradient Projection Descent

Notice that a new gradient from point C can be determined and the tip of the vector CD projected to the boundary as before. This time movement from C to E is along the boundary. This movement along the boundary leads one to the conclusion that for the orientation of the control boundaries on the contour map of Figure 4a, the driving function will always go to the corner of the boundaries. The termination of the driving function $\underline{u}$ at the corner of the boundaries is in keeping with the maximum principal of Pontriagin which states that the magnitude of the driving function $\underline{u}$ must be the maximum allowed by the physical limitations of the system in order to achieve time optimization. A second descent scheme, the Corner Aiming Descent Scheme, can be formulated by taking advantage of this principal. Figure 4b illustrates this second descent scheme geometrically.

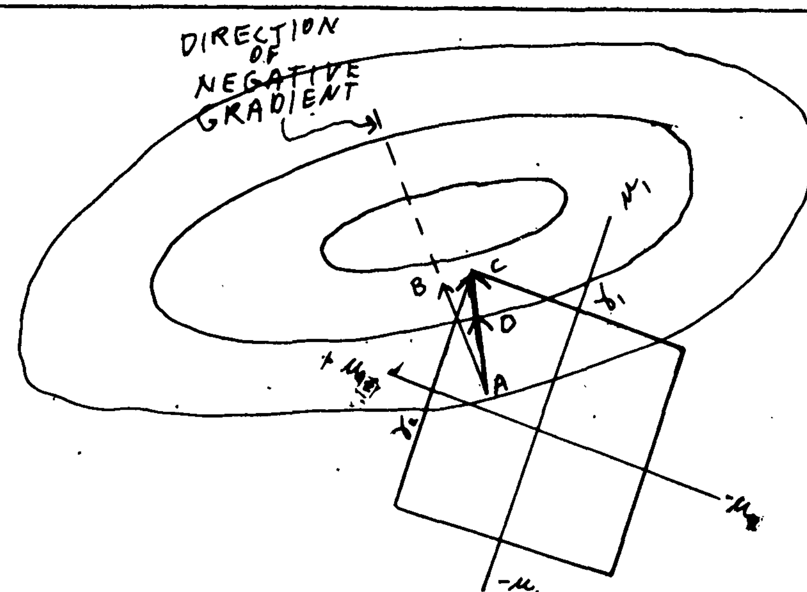

Figure 4b
Geometrical Interpretation of Corner Aiming Descent

After the gradient is determined, instead of projecting the tip of the vector $\eta \underline{g}$ down to the boundary, the corner nearest the vector AB is aimed for. In this way, the assumption that the corner will eventually be reached is utilized to aid in the descent. The use of the maximum principal thus enters into the calculations.

All of the statements made about the two dimensional models of Figures 4a and 4b can of course be extended to K dimensional hyperspace for the K number of control subintervals encountered in an actual problem. The mathematics for computing the change in the driving function $\delta \underline{u}$ utilizing these two descent schemes is presented below.

The following equations describe the Gradient Projection Descent Scheme:

$$w_k = \begin{cases} \gamma_h - u_k & \text{IF} \quad \delta u_k \geq \gamma_h - u_k \\ \delta u_k & \text{IF} \quad \gamma_l - u_k < \delta u_k < \gamma_h - u_k \\ \gamma_l - u_k & \text{IF} \quad \delta u_k \leq \gamma_l - u_k \end{cases} \qquad k = 0, \cdots, K-1 \tag{114}$$

where γ_h is the upper boundary for the driving function and γ_l is the lower boundary for the driving function. A new η must now be determined in the same manner as before:

$$\eta_1 = \frac{-\underline{g}'\underline{w}}{2\underline{w}'\underline{H}'\underline{R}\,\underline{H}\,\underline{w}} \tag{115}$$

It is now necessary to assure that η_1 does not exceed unity:

Therefore

$$\eta_2 = sat(\eta_1) \tag{116}$$

where

$$\begin{aligned} sat(x) &= x \quad \text{IF } |x| \leq 1 \\ &= \pm 1 \quad \text{IF } |x| > 1 \end{aligned} \tag{117}$$

Finally, the change in the driving function is expressed as

$$\delta \underline{u} = \eta_2 \underline{w} \tag{118}$$

Conceptually, the above procedure limits $\underline{w}$ as shown in Figure 5, below.

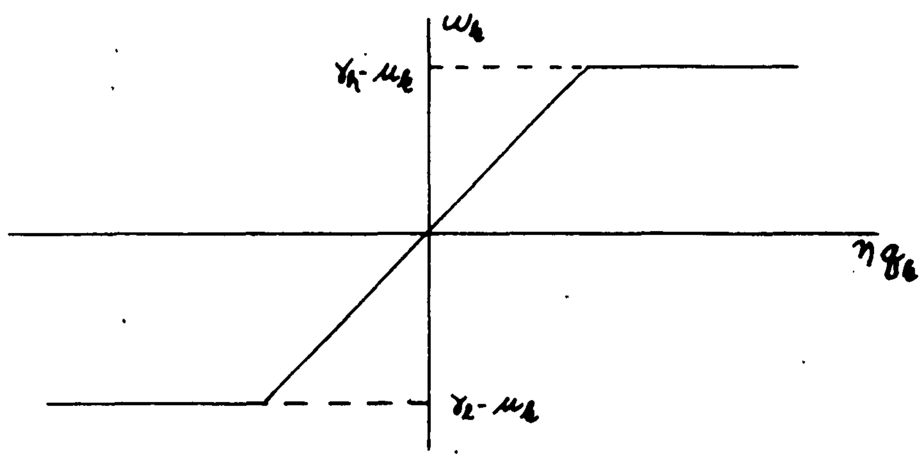

Figure 5
Gradient Projection Descent

The following equations formulate the Corner Aiming Descent Scheme:

$$w_k = \begin{cases} \gamma_h - u_k & \text{IF} \quad g_k < 0 \\ 0 & \text{IF} \quad g_k = 0 \\ \gamma_\ell - u_k & \text{IF} \quad g_k > 0 \end{cases} \tag{119}$$

As before it is necessary to determine a new η:

$$\eta_1 = \frac{-\underline{g}'\underline{w}}{\underline{w}'\underline{H}'\underline{R}\,\underline{H}\,\underline{w}} \tag{120}$$

Again η_1 must be limited to unity:

$$\eta_2 = sat\,(\eta_1) \tag{121}$$

Finally, as before

$$\delta\underline{u} = \eta_2\,\underline{w} \tag{122}$$

In the Corner Aiming Descent Scheme w is limited in a different way. Figure 6 on the next page shows this conceptually.

The new value of the driving function can now be computed by adding the value for $\delta\underline{u}$ computed by either the Gradient Projection Descent Scheme or the Corner Aiming Descent Scheme to the old value of $\underline{u}$. This is expressed mathematically

$$\underline{u}_{new} = \underline{u}_{old} + \delta\underline{u} \tag{123}$$

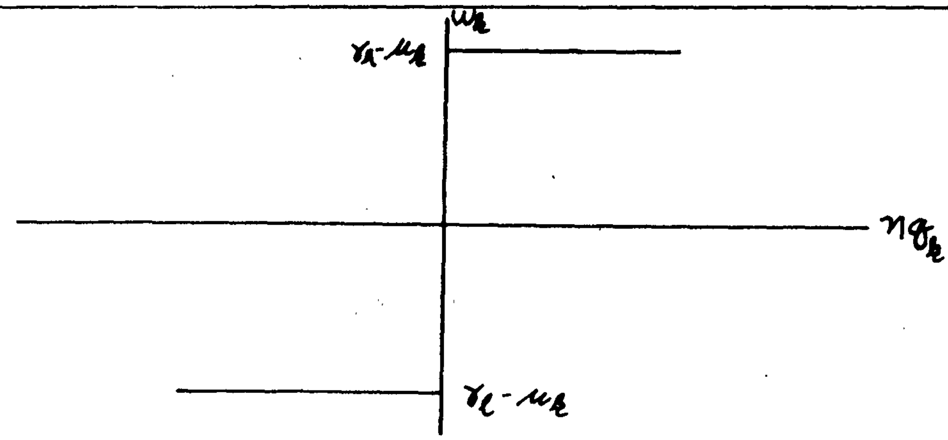

Figure 6

Corner Aiming Descent

Completion of Solution

The process is now repeated with the new value of $\underline{u}$ and the solution will hopefully converge to a true minimum. It is necessary, however, to determine when the minimum has been reached. In the method of Steepest Descent a minimum is reached as soon as the gradient becomes zero as in Figure 6a on the next page. However, in this problem the constraint on the driving function may prevent all the elements of the gradient vector from reaching zero. Therefore, conditions other than a zero gradient will have to be used to determine when the minimum for the terminal norm V has been reached. The solution

is considered complete whenever one of the three conditions below occurs for all u_k:

$$
\begin{aligned}
&\gamma_h - u_k = 0 && \text{IF} \quad g_k < 0 \\
&\gamma_l < u_k < \gamma_h && \text{IF} \quad g_k = 0 \\
&\gamma_l - u_k = 0 && \text{IF} \quad g_k > 0
\end{aligned}
\qquad (124)
$$

Equation (124) states that the solution is complete when, for each control subinterval, 1) the gradient has reached zero and no further improvement in the terminal norm V can be made, or 2) that no further change is possible in the driving function u_k due to the constraint imposed. When one of these conditions is met for every

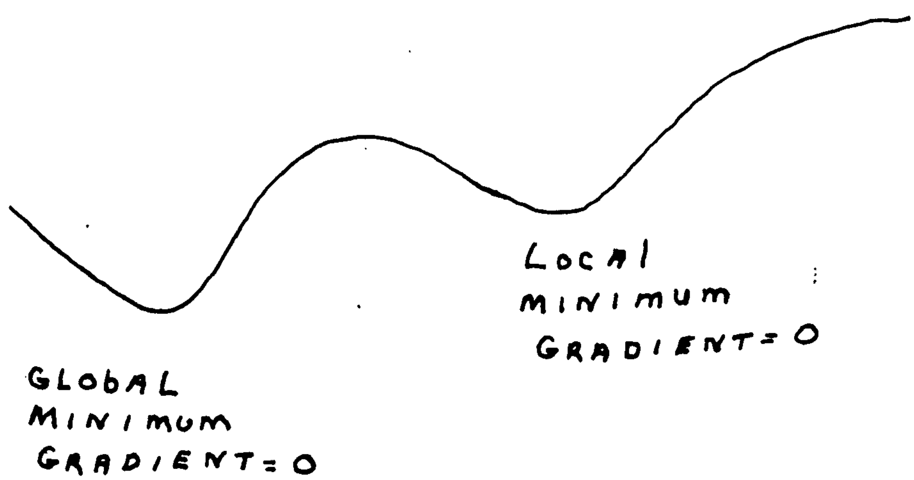

Figure 6a
Local and Global Minima

control subinterval computation is halted and the final driving function is read out. That the point reached is a minimum is shown in Reference 5 (Ref 5:A-20toA-23).

It should be noted that any time the method of Steepest Descent is employed the minimum reached may only be a local minimum. There may still exist someplace on the hypersurface considered a global minimum. Figure 6a illustrates this for a two dimensional problem. This should not detract from the usefulness of this method, however, since, for the type of problem being considered in this report, the approximate solution is already known from the maximum principal. According to the maximum principal the solution is expected to be "bang-bang" and thus an approximate check on the results is available.

This completes the mathematical development for the solution for the optimum driving function of a linear controller. The next chapter of this report will present a description of the computer programs used and a general discussion of their use.

IV. Description of the Computer Programs

The purpose of this chapter is to extract from Chapter III the various equations utilized in the digital computer programs and present them in a meaningful manner; in addition, certain heretofore unexplained phases of the solution will be treated in detail. Since the development of the equations of the descent has been set forth in great detail in the preceding chapter, the equations extracted from that chapter will be stated without further justification. Appendix A will, by reference to the listings of the programs, explain the purpose of the various FORTRAN statements used.

Computation of $\underline{p}$, $\underline{a}$, and $\underline{H}$

Although $\underline{p}$, $\underline{a}$, and $\underline{H}$ were defined in Chapter III the details of their computation remain to be discussed. Recalling the definitions of $\underline{p}$, $\underline{a}$, and $\underline{H}$

$$\underline{p} = e^{\underline{F}T}\underline{c} = \underline{\Phi}(T)\,\underline{c} \tag{95}$$

$$\underline{a} = \int_0^{\tau} \underline{\Phi}(\sigma)\,\underline{g}\,d\sigma \tag{93}$$

$$\underline{H} = \left[\underline{\Phi}(T-\tau)\,\underline{a} \quad \underline{\Phi}(T-2\tau)\,\underline{a} \quad \cdots \quad \underline{a}\right] \tag{94}$$

The most obvious way to compute these functions is to use the definition of $\underline{\Phi}$:

$$\underline{\Phi}(t) = e^{\underline{F}t} \; ; \quad \underline{\Phi}(0) = \underline{I} \tag{78}$$

Equation (78) can easily be calculated by using the definition of $e^{\underline{F}t}$:

$$e^{\underline{F}t} = \sum_{n=0}^{\infty} \frac{\underline{F}^n t^n}{n!} \quad ; \quad \underline{F}^0 = \underline{I} \tag{79}$$

The successive terms of the series are computed and summed on a digital computer for a particular value of t. This method was originally used for computing $\underline{\Phi}$ and worked quite well for problems with a short control interval and small moduli of the elements in the matrix $\underline{F}$. Due to the arithmetical nature of the digital computer it is necessary to solve the numerical rather than the algebraic problem. For instance, for each column of $\underline{H}$ a seperate $\underline{\Phi}(T-\Delta\tau)$ must be computed.

The number of terms in the series for $\underline{\Phi}$ necessary for the required accuracy varies with the moduli of the elements in the numerator of the terms of the series. (Note that for the actual computation the matrix $\underline{F}$ is multiplied through by the scalar t before the numerator is carried to a power.) The size of the numerator moduli depends not only on the elements of the matrix but also on the value chosen for t. Experience has shown that whenever it is necessary to carry the series farther than 100 terms the computation time is excessive and roundoff errors in the computer arithmetic cause inaccuracies. It should be noted, however, that the series method is a straightforeward and quite practical method for computing $\underline{\Phi}$ provided the moduli of the numerator of the first term of the series do not exceed 25.

The accuracy of the result may be assured in either one of two ways. Each term computed may be examined and whenever the absolute value of each element of the matrix of that term is less than the required accuracy computation may be considered complete. A less difficult method from the standpoint of programming is the use of a procedure set forth in Section 4-19 of Elementary Matrices (Ref 3:145). The upper bound for the power of a matrix is first computed by this procedure. The result is then divided by $m!$. One thus computes the upper bound of the matrix $\frac{\underline{F}^m \chi^m}{m!}$ for each term of the series and compares it with the accuracy required. When the upper bound falls below the required accuracy the number of the term just calculated is read out and employed to terminate the rest of the programs involving series computations.

Computation of the fundamental matrix for a problem of a practical nature is usually not feasible by the series solution method because of the large number of terms necessary for accuracy. However, another method involving direct numerical integration of the system equations can be employed.

Computation of $\underline{P}$, $\underline{a}$, and $\underline{H}$ by the Method of Runge-Kutta

The matrices $\underline{p}$, $\underline{a}$, and $\underline{H}$ can be computed by direct integration of the system equations. It is the purpose of this section to develop the theory for this method. Consider the solution to the system equations:

$$\underline{x}(T) = \underline{\Phi}(T)\underline{c} + \int_0^T \underline{\Phi}(T,\sigma)\, \underline{g}\, u(\sigma)\, d\sigma \tag{72}$$

Examination of equation (72) reveals that in the linear system the solution is constructed by superposition of the response due to the initial condition $\underline{x}(0) = \underline{c}$ and the response due to the input $u(\sigma)$. Taking first of all

$$\underline{p} = \underline{\Phi}(T)\,\underline{c} \tag{95}$$

one notices that this could be regarded as the solution

$$\underline{p} = \underline{x}(T) = \underline{\Phi}(T)\,\underline{c} + \underline{0} \tag{125}$$

Equation (125) is identical to the solution for the unexcited system with an initial condition

$$\underline{x}(0) = \underline{c} \tag{126}$$

The term "unexcited" means that the driving function $u(\sigma)$ is zero in equation (72). Equation (125) is thus the solution to the homogeneous system

$$\dot{\underline{x}}(t) = \underline{F}\,\underline{x}(t) \quad ; \quad \underline{x}(0) = \underline{c} \tag{127}$$

at $t = T$. Direct integration of equation (127) will then determine $\underline{p}$.

The solution for the vector $\underline{a}$ will be considered next. Taking the integral

$$\underline{a} = \int_0^T \underline{\Phi}(\sigma)\,\underline{g}\,d\sigma \tag{93}$$

and carrying out the integration as before, one has

$$\int_0^{\tau} \underline{\Phi}(\sigma)\, \underline{g}\, d\sigma = \underline{F}^{-1}\left[e^{\tau \underline{F}} - \underline{I}\right] \underline{g} \tag{128}$$

Now take the integral

$$\int_0^{\tau} \underline{\Phi}(\tau-\sigma)\, \underline{g}\, d\sigma = \int_0^{\tau} e^{(\tau-\sigma)\underline{F}}\, \underline{g}\, d\sigma \tag{129}$$

and carry out the integration:

$$e^{\tau \underline{F}} \int_0^{\tau} e^{-\sigma \underline{F}}\, \underline{g}\, d\sigma = -\underline{F}^{-1} e^{\tau \underline{E}} \left[e^{-\sigma \underline{F}}\right]_0^{\tau} \underline{g} \tag{130}$$

$$= \underline{F}^{-1}\left[e^{\tau \underline{F}} - \underline{I}\right] \underline{g}$$

The results of equations (128) and (130) are identical. The original integrals may thus be equated:

$$\underline{a} = \int_0^{\tau} \underline{\Phi}(\sigma)\, \underline{g}\, d\sigma = \int_0^{\tau} \underline{\Phi}(\tau-\sigma)\, \underline{g}\, d\sigma \tag{131}$$

Referring back to the solution to the system equations, which is equation (72) one sees that

$$\underline{a} = \underline{x}(\tau) = \underline{0} + \int_0^{\tau} \underline{\Phi}(\tau-\sigma)\, \underline{g}\, (1)\, d\sigma \tag{132}$$

Equation (132) says that the vector $\underline{a}$ may be calculated by finding the solution at time $t=\tau$ to the system equations which have been excited with a step input of unity with an initial condition $\underline{x}(0) = \underline{0}$.

Thus the system

$$\dot{\underline{x}}(t) = \underline{F}\, \underline{x}(t) + \underline{g}(1) \quad ; \quad \underline{x}(0) = \underline{0} \tag{133}$$

may be integrated from $t = 0$ to τ to determine $\underline{a}$.

Finally, one can compute the matrix $\underline{H}$ by noting that

$$h_{ik} = \varphi_{ij}(T - k\tau)a_j \qquad k = 0, \cdots, K-1 \tag{134}$$

This can be expressed alternately as

$$\underline{h}_k = \underline{x}(T - k\tau) = \underline{x}(m\tau) = \underline{\Phi}(m\tau)\underline{a} + \underline{0} \tag{135}$$
$$m = (K-1-k) \qquad k = 0, \cdots, K-1$$

Thus the columns of the $\underline{H}$ matrix can be calculated by computing the state vector of the unexcited system at the control subintervals starting with an initial condition $\underline{x}(0) = \underline{a}$. One then must evaluate the following system at $t = 0$, then integrate from $t = 0$ to τ, from $t = \tau$ to 2τ, etc.

$$\dot{\underline{x}}(t) = \underline{F}\,\underline{x}(t) \quad ; \quad \underline{x}(0) = \underline{a} \tag{136}$$

Now that formulae which can be directly integrated to find $\underline{p}$, $\underline{a}$, and $\underline{H}$ have been developed, a practical method of integration must be considered.

Again, the only practical method of integration is an approximate numerical solution. Of the many methods available, the method of Runge-Kutta is quite suitable for this type of problem since it affords accurate results and since the computation procedures are recurrent. Also, it may be extended quite easily from the scalar form to the matrix form. Consider one equation of a State Space system in subscripted form:

$$\frac{dx_1}{dt} = f_{11}x_1 + f_{12}x_2 + \cdots + f_{1n}x_n + g_1 \tag{137}$$

Replace dt by a small interval Δt and apply the following formulae(Ref 10: 123):

if $\quad \dot{x}(t) = f(x,t)\ ; \quad x(0) = c$

then

$$
\begin{aligned}
dx_1 &= f(c,0)\,\Delta t \\
dx_2 &= f\left(c+\frac{dx_1}{2}, \frac{\Delta t}{2}\right)\Delta t \\
dx_3 &= f\left(c+\frac{dx_2}{2}, \frac{\Delta t}{2}\right)\Delta t \\
dx_4 &= f(c+dx_3, \Delta t)\,\Delta t
\end{aligned}
\tag{138}
$$

$$\Delta x = \frac{1}{6}(dx_1 + 2dx_2 + 2dx_3 + dx_4)$$

The formulae may in like manner be applied to the successive State equations to compute $\Delta x_1 \cdots \Delta x_8$ In matrix form this is more compactly expressed for the particular problem of calculating $\underline{a}$:

$$
\begin{aligned}
\underline{x}(0) &= \underline{c} = \underline{0} \\
\underline{dx}_1 &= \underline{F}\,\underline{x}\,\Delta t + \underline{g}\,\Delta t \\
\underline{dx}_2 &= \underline{F}\left(\underline{x} + \tfrac{1}{2}\underline{dx}_1\right) + \underline{g}\,\Delta t \\
\underline{dx}_3 &= \underline{F}\left(\underline{x} + \tfrac{1}{2}\underline{dx}_2\right) + \underline{g}\,\Delta t \\
\underline{dx}_4 &= \underline{F}\left(\underline{x} + \underline{dx}_3\right) + \underline{g}\,\Delta t \\
\Delta\underline{x} &= \tfrac{1}{6}\left(\underline{dx}_1 + 2\underline{dx}_2 + 2\underline{dx}_3 + \underline{dx}_4\right)
\end{aligned}
\tag{139}
$$

The value of $\Delta \underline{x}$ just calculated is then added to $\underline{x}(0)$ and the new value of $\underline{x}$ is used as an initial condition for another round of computations. The computation thus proceeds in an iterative fashion until the particular value of t is reached. The value of $\underline{x}$ at this point is then read out as the desired solution.

It is possible, using the method of Runge-Kutta just described, to compute $\underline{P}$, $\underline{a}$, and $\underline{H}$ by direct integration of the system equations. The utility of this method is obvious in that it does not depend on a series solution and hence the accuracy is not limited by the number of terms involved. Thus, a problem for which the moduli of the matrix $\underline{F}$ are large may be handled as easily as any other. Also, when computing $\underline{H}$, it is only necessary to read out the solution at the end of each successive control subinterval rather than to start from the beginning for each column as with the series solution. One must only be careful to choose a proper value for Δt. Too small a Δt will result in a roundoff error when using a digital computer because the $\Delta \underline{x}$ computed will be quite small. Conversely, when Δt is too large the accuracy will suffer because the slope of the function will not remain constant over the interval of Δt.

Calculation of $\underline{P}$, $\underline{a}$, and $\underline{H}$ represents precomputation which is accomplished before the main program, which is the descent, is entered. This not only breaks up the computation time, but also allows one to check for reasonable results of the precomputation before employing the descent program.

The Descent

Now that the precomputation has been explained, it is only necessary to list the equations of the descent in the order in which they are programmed. It should be noted that an initial value for the driving function $\underline{u}$ must be guessed at and a control interval T selected. It goes without saying that a good initial guess for $\underline{u}$ will greatly speed the solution. The choice of the control interval will of course depend on the linear transfer function itself, on the limits of the driving function, on the initial conditions, and on the $\underline{R}$ matrix. Choice of the $\underline{R}$ matrix, which is a matrix of weighting constants, depends on the amount of direct control it is desired to exert on each state variable. Further discussion of the choice of the matrix $\underline{R}$ will be postponed until the end of this chapter.

Using the precomputed values for $\underline{p}$ and $\underline{H}$, and a selected $\underline{R}$ the following equations comprise the descent portion of the solution:

$$\underline{x}(T) = \underline{p} + \underline{H}\,\underline{u} \tag{96}$$

$$V = \underline{x}'(T)\,\underline{R}\,\underline{x}(T) \tag{61}$$

$$\underline{g} = \underline{H}'\,\underline{R}\,\underline{x}(T) \tag{140}$$

$$\eta = \frac{-\underline{g}'\underline{g}}{\underline{g}'\,\underline{H}'\,\underline{R}\,\underline{H}\,\underline{g}} \tag{141}$$

$$\delta\underline{u} = \eta\,\underline{g} \tag{110}$$

The operator must now choose between the two descent schemes. The equations for the Gradient Projection Descent Scheme are presented first:

$$w_k = \begin{cases} \gamma_h - u_k & \text{IF} \quad \delta u_k \geq (\gamma_h - u_k) \\ \delta u_k & \text{IF} \quad (\gamma_l - u_k) < \delta u_k < (\gamma_h - u_k) \\ \gamma_l - u_k & \text{IF} \quad \delta u_k \leq (\gamma_l - u_k) \end{cases} \tag{114}$$

A new value of η must now be computed:

$$\eta_1 = \frac{-\underline{g}'\underline{w}}{\underline{w}'\underline{H}'\underline{R}\,\underline{H}\,\underline{w}} \tag{142}$$

The above is then limited to unity:

$$\eta_2 = sat(\eta_1) \tag{116}$$

The change in the driving function is calculated next:

$$\delta\underline{u} = \eta_2\,\underline{w} \tag{118}$$

Finally

$$\underline{u}_{new} = \underline{u}_{old} + \delta\underline{u} \tag{123}$$

If the operator desires, the Corner Aiming Descent Scheme may be used:

$$w_k = \begin{cases} \gamma_h - u_k & \text{IF} \quad g_k < 0 \\ 0 & \text{IF} \quad g_k = 0 \\ \gamma_l - u_k & \text{IF} \quad g_k > 0 \end{cases} \tag{119}$$

Again a new value for η must be calculated:

$$\eta_1 = \frac{-\underline{g}'\underline{w}}{\underline{w}'\underline{H}'\underline{R}\,\underline{H}\,\underline{w}} \tag{143}$$

and

$$\eta_2 = sat(\eta_1) \tag{121}$$

From which

$$\delta\underline{u} = \eta_2\,\underline{w} \tag{122}$$

Finally

$$\underline{u}_{new} = \underline{u}_{old} + \delta\underline{u} \tag{123}$$

The computation is repeated with the new value of $\underline{u}$. The process is iterative until the following conditions are fulfilled for all control subintervals, whereupon it is stopped and the final control is read out:

$$\begin{array}{ll} |\gamma_h - u_k| \leq \delta & \text{FOR} \quad g_k < -\varepsilon \\ \gamma_\ell < u_k < \gamma_h & \text{FOR} \quad |g_k| \leq \varepsilon \\ |\gamma_\ell - u_k| \leq \delta & \text{FOR} \quad g_k > \varepsilon \end{array} \tag{144}$$

The terms δ and ε must be used in a digital computer solution since the control converges toward a boundary, but never quite reaches it. Also, the gradient never quite becomes zero, although it approaches it so closely that no noticeable improvement can be

made in the terminal norm. Thus, further computation would be wasted effort.

The Matrix $\underline{R}$

A brief discussion of the choice of the matrix $\underline{R}$ is now in order. The function

$$V = \underline{x}'(T)\ \underline{R}\ \underline{x}(T) \tag{61}$$

is termed a "quadratic" if $\underline{R}$ is symmetrical. When this is expanded one has

$$V = r_{11}\, x_1^2(T) + r_{22}\, x_2^2(T) + \cdots + r_{nn}\, x_n^2(T) + 2 r_{ij}\, x_i(T)\, x_j(T) \tag{145}$$

The ij terms are called mutual dependance terms, while the ii terms are called self dependance terms. When formulating the criterion V, it is desirable to keep only the self dependance terms, since one is usually not interested in controlling the product of two state variables. Thus, the r_{ij} are made zero. The r_{ii} then become weighting constants which reflect the "utility" of controlling directly each state variable.The term "directly" means that an attempt is made to drive that particular state variable to zero during the control interval. This becomes more clear when one notes that for any r_{ii} chosen to be zero the corresponding x_i will not be driven to zero but will follow according to the equation

$$x_i = \frac{dx_{i-1}}{dt} - g_{i-1}\, u \tag{146}$$

With these points in mind the engineer may choose the elements of the matrix β in order to solve the particular problem he is confronted with.

As in all minimization or maximization problems the choice of the criterion used influences the answer a great deal. Since the matrix $\underline{\beta}$ is a part of the criterion used for the particular method discussed in this report it will certainly influence the answer a great deal, as will be seen in the next chapter.

Determining the Optimum Control Interval

After the iteration process has been completed and the optimum driving function has been calculated for a particular control interval, the computed value for the terminal norm V is read out. The process is then repeated for a new control interval. After this is done several times, the values of V are plotted against the respective control intervals used for the computations. The result is that shown in Figure 7 on the next page. The end of the solid part of the curve represents the last value of V plotted. A tangent to the curve is then projected from this point to the T axis as shown. The value of T at the intercept of the tangent with the T axis can then be used as an estimate for a new control interval for the next computation. The process may then be repeated successively. Note that the dashed portion of the V curve is assymtotic to the T axis. Since the curve never reaches zero, a small percentage of the initial value for V is picked. This value is plotted on the V axis and a horizontal line

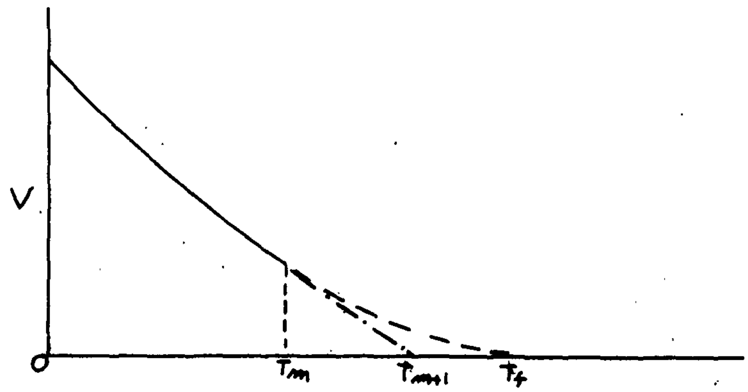

Figure 7
Determination of New Control Interval

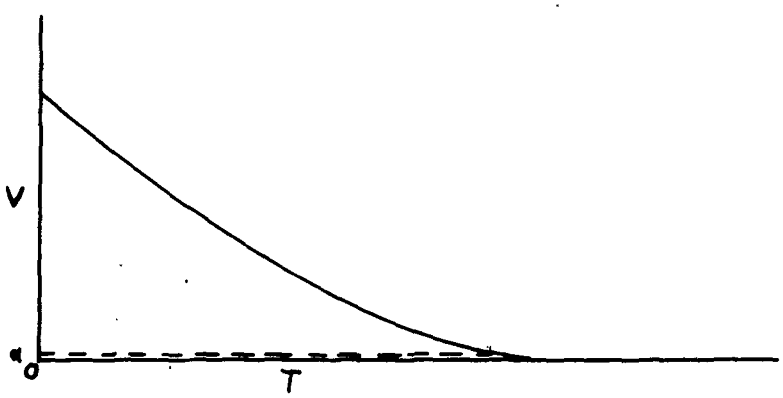

Figure 8
Terminal Norm vs. Control Interval

is drawn as shown in Figure 8. When the value of V just falls below this line the corresponding T is the optimum control interval and the driving function computed at this value of T is the optimum driving function.

This completes the description of the computer programs. The next chapter of this report will be devoted to the analysis of the Minneapolis-Honeywell relay servo loop.

V. State Space Analysis of the Minneapolis-Honeywell Relay Servo Loop

In this part of the report the numerical solutions for the optimum driving function of the linear controller portion of the Minneapolis-Honeywell Adaptive Control System will be discussed. In Chapter II it was stated that only the relay servo loop of the Honeywell system would be analyzed in this report, since the model could easily be analyzed seperately. A block diagram of the relay servo loop is presented on the next page in Figure 9.

A meaningful method of analysis for any servo system is to find the response to a unit step input; this method will be employed in this report. In Chapter III a method for finding the optimum driving function to drive an initial error to zero in minimum time was developed for a linear controller. If the relay servo loop of Figure 9 is excited by a unit step input, it will appear at the output of the summer as a unit step error, provided the system was in equilibrium to start with. The switching logic and the relay will then generate a driving function for the linear controller to drive the error back to zero. Equivalent to a unit step input with an initial output error of zero would be a zero input with an initial output error of a unit step. The output of the summer would be the same in both cases disregarding the signs. Since the two are equivalent, the problem that will be considered in this chapter is that of computing the optimum driving function for the linear controller portion of the Honeywell relay servo loop ~~when~~ the system has an initial error

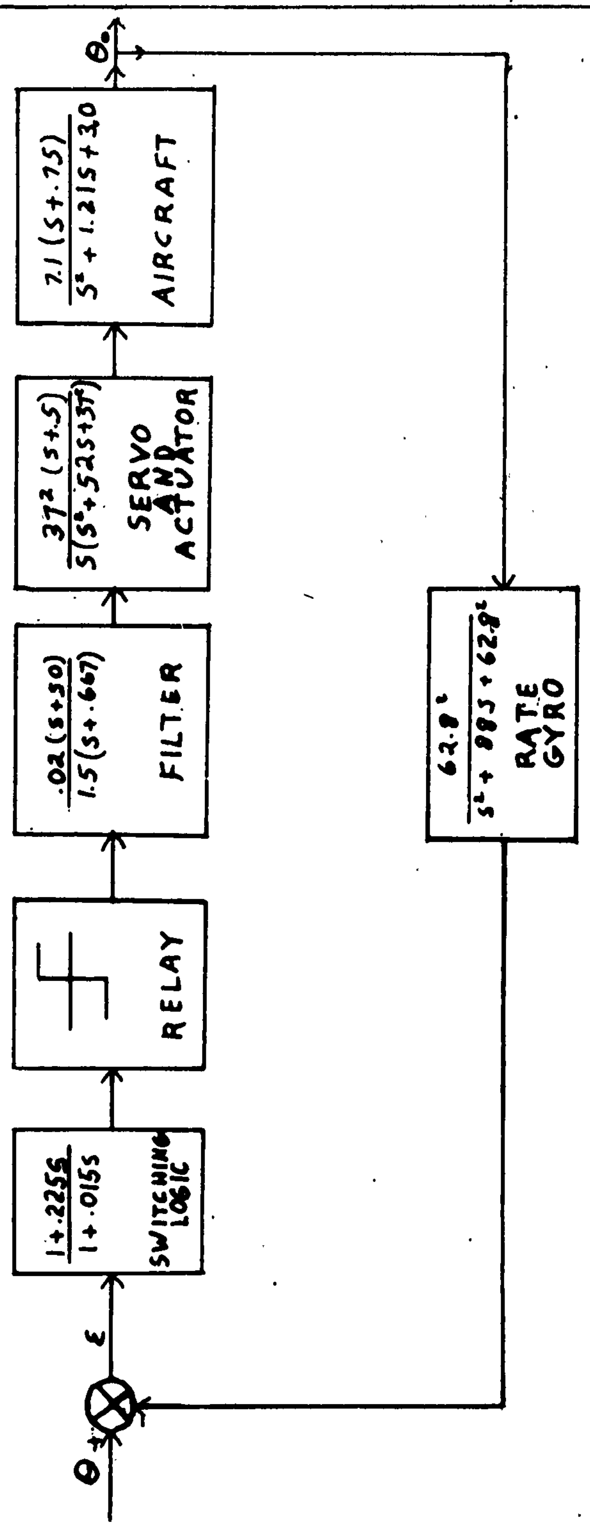

Figure 9
Relay Servo Loop Used in Analysis

of a unit step.

The purpose of the analysis of this chapter is twofold: first, to obtain an optimum driving function for the linear portion of the controller which can be compared to the relay output of the Honeywell relay servo loop as observed during the analogue computer simulation; second, to demonstrate the capabilities and limitations of the State Space techniques and the method of Steepest Descent.

Solutions were attempted to five problems. Two of these problems involved a fourth order approximation of the sixth order linear controller. The five solutions sought were: 1) a solution for direct control of all six state variables of the sixth order system, 2) a solution for direct control of the first four state variables of the sixth order system, 3) a solution to control directly all four state variables of the fourth order approximation, 4) a solution to control directly only the first two state variables of the sixth order system, and 5) a solution to control directly only the first two state variables of the fourth order approximation. The fourth order approximation was employed after numerical analysis of the sixth order system indicated that the descent would probably be slow to converge, a fear that was later borne out.

The Fourth Order Approximation

The transfer function for the sixth order linear controller is

$$\frac{e_o}{e_{in}} = \frac{195(S+50)(S+.75)(S+.5)}{S(S+.667)(S^2+52S+37^2)(S^2+1.21S+3.0)} \tag{147}$$

Examination of equation (147) indicates that a good approximation to the system could be made by eliminating the high frequency poles and zero. The result is the transfer function

$$\frac{e_o}{e_{in}} = \frac{7.1\,(S+.75)\,(S+.5)}{S(S+.667)(S^2+1.21S+3.0)} \tag{148}$$

Also, compare Figure 10 with Figure 11. The former is the Log Magnitude and Phase Angle Diagram of the sixth order transfer function, while the latter is the Log Magnitude and Phase Angle Diagram for the fourth order transfer function. Note that the phase margin of 45^o occurs at about the same frequency on both diagrams as does the gain crossover. The fourth order transfer function thus represents a reasonable approximation of the Honeywell linear controller at the frequencies which will be pertinent to this analysis.

<u>The Minneapolis-Honeywell Switching Logic</u>

In this section the Honeywell switching logic will be analyzed in order to determine a set of weighting constants to be used in the matrix $\underline{R}$. The transfer function of the switching logic is

$$\frac{e_o}{\mathcal{E}} = \frac{1+.225\,S}{1+.015\,S} \tag{149}$$

If this transfer function is excited by a unit ramp input $\frac{1}{S^2}$ the output is essentially proportional plus derivative. For

$$\mathcal{E}(s) = \frac{1}{S^2} \tag{150}$$

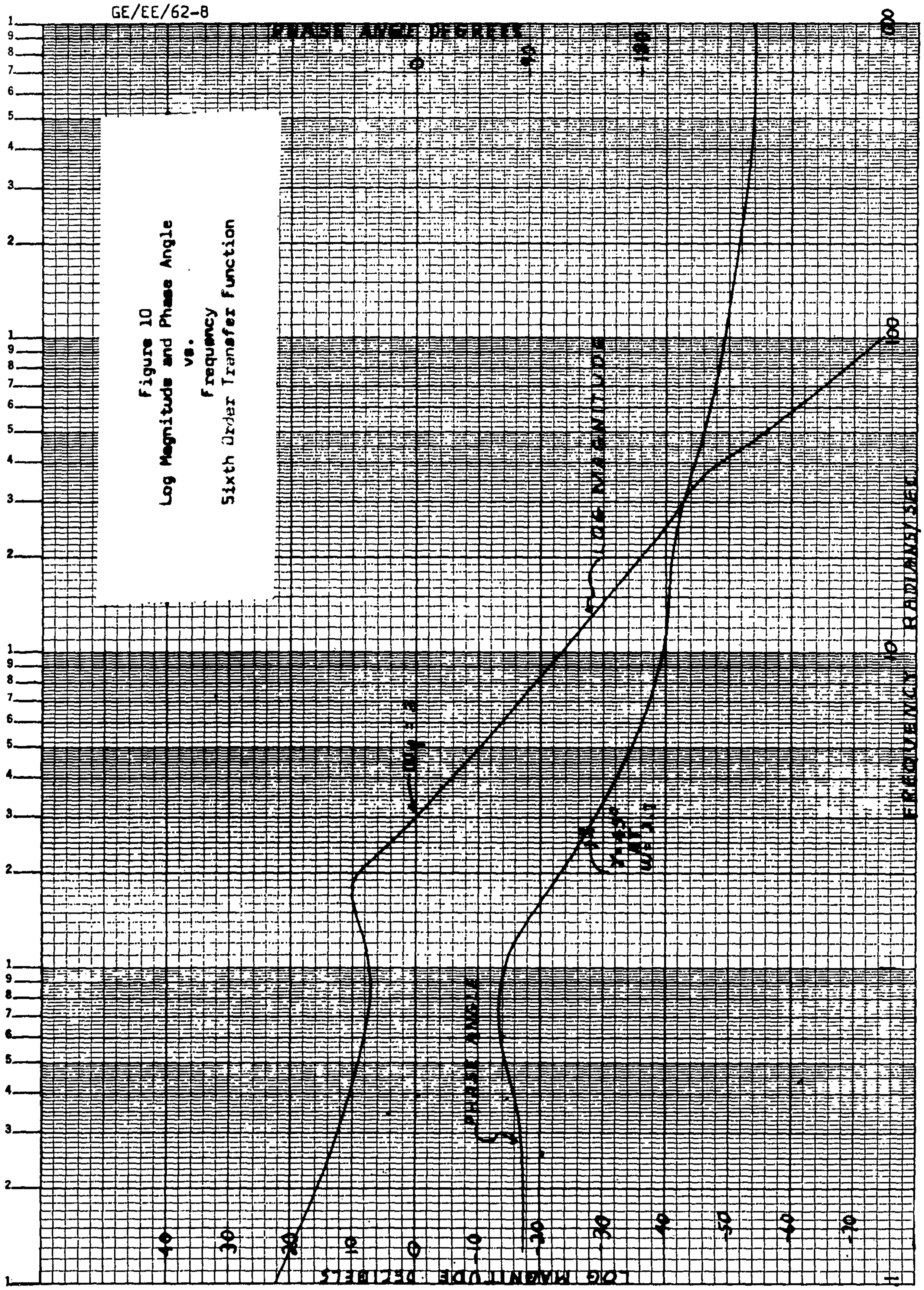

Figure 10
Log Magnitude and Phase Angle
vs.
Frequency
Sixth Order Transfer Function

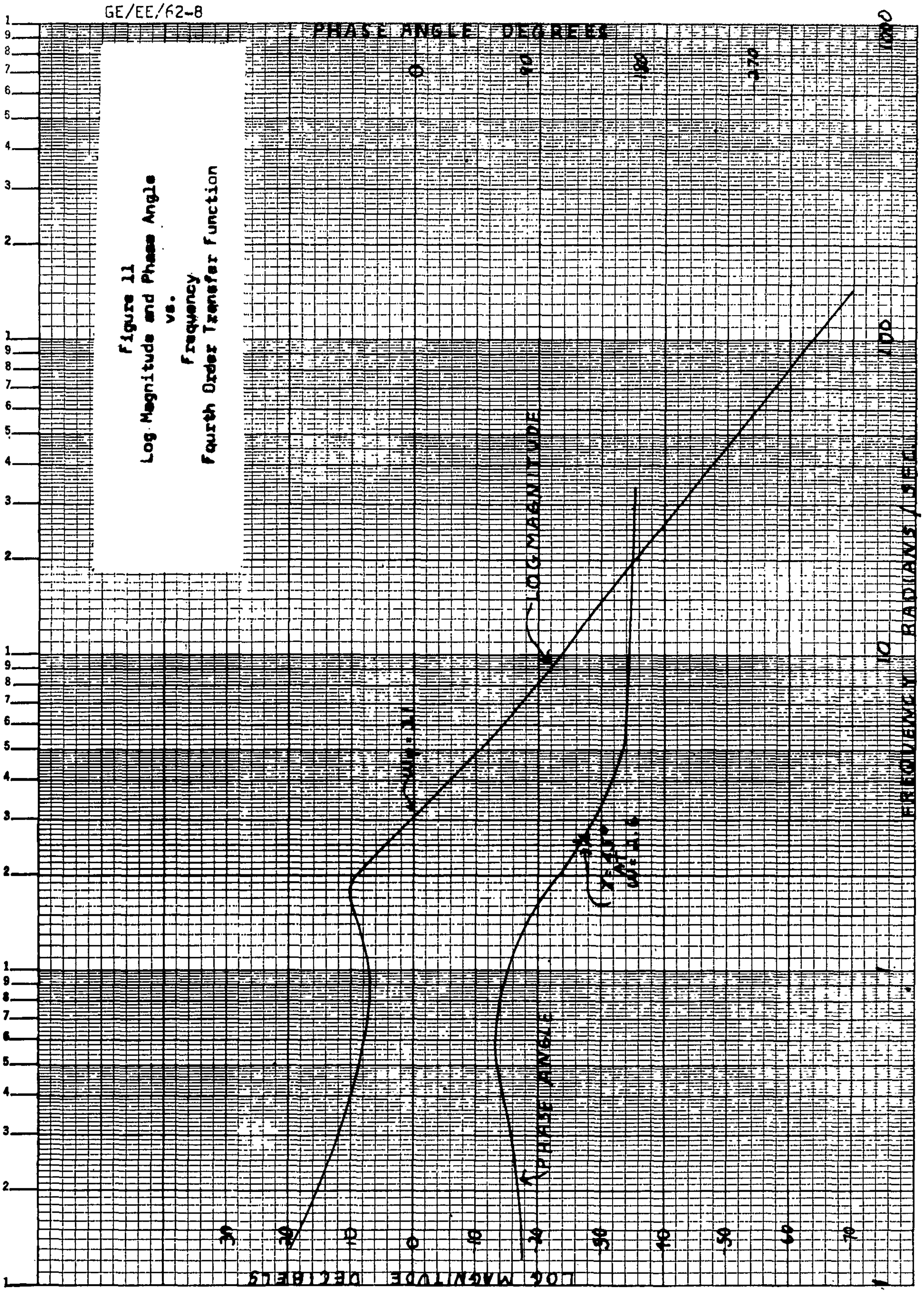

Figure 11
Log Magnitude and Phase Angle
vs.
Frequency
Fourth Order Transfer Function

the output in the time domain becomes approximately

$$e_o(t) \approx t + .225 \tag{151}$$

When the transfer function for the linear controller portion of the Honeywell relay servo loop is expressed in State Space notation $x_2 = \frac{dx_1}{dt}$ where x_1 is the system error. The switching logic then forms the sum $x_1 + .225\,x_2$ when x_1 alone is applied to the switching logic. Note that when driving the controller using the Honeywell switching logic only the first two state variables are measured and used as a basis for direct control. The other four state variables are not directly controlled, but follow according to the equation

$$x_i = \frac{dx_{i-1}}{dt} + g_{i-1}\,u \tag{146}$$

Finally, viewing the switching logic from the standpoint of formulating a terminal norm, one could say that weights assigned to x_1 and x_2 for direct control are 1 and .225 respectively.

Choice of the Matrix $\underline{R}$

In Chapter IV it was stated that the choice of the matrix $\underline{R}$ depended on the "utility" of controlling the various state variables directly. Admittedly the Honeywell switching logic did not employ the quadratic criterion function, but it rather summed algebraicly the two weighted state variables. Nevertheless, to form a basis for comparison, the r_{11} and the r_{22} of the $\underline{R}$ matrix were chosen

to be 5 and 1.12 respectively. The rest of the $\underline{R}$ matrix was chosen to be zero. Two problems were solved using this matrix for $\underline{R}$: one for the sixth order transfer function, and one for the fourth order approximation.

In order to demonstrate the limitations and capabilities of the methods used, three other problems were formulated. Two solutions to the sixth order problem with $\underline{R}$ diagonal were attempted. In the first case the diagonal elements were 6, 5, 4, 3, 2, and 1 starting with r_{11} ; in the second case they were 4, 3, 2, 1, 0, and 0. Another solution which was attempted was one to the fourth order approximation with $\underline{R}$ diagonal; the diagonal elements were 4, 3, 2, and 1.

<u>Data for the Solution</u>

The sixth and fourth order transfer functions were put into State Space form by means of the linear transformations developed in Chapter III. Tables I and II on the following two pages show the data used in the computer solutions for the sixth and fourth order problems respectively. The number of control subintervals, and hence the lengths of the control intervals used were varied.

Note the magnitude of the last two elements of the $\underline{g}$ vector in the sixth order problem. This is due to the nature of the recurrence formula

$$G_i = b_i - \sum_{k=0}^{i-1} a_{i-k} G_k \qquad (32)$$

Table I

Data for Sixth Order Problem

$$\frac{e_o}{e_{in}} = \frac{195\,(s^3 + 51.25 s^2 + 62.875 s + 18.75)}{s(s^5 + 53.877 s^4 + 1970.41 s^3 + 2769.8 s^2 + 5320 s + 2738)}$$

$$\underline{F} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -2738 & -5370 & -2769.8 & -1970.41 & -53.877 \end{bmatrix}$$

$$\underline{g} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -2.627 \\ -1264.0 \\ 69{,}214.0 \end{bmatrix}$$

$$\underline{x}(0) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

TAU = .025

UHI = 195.0

ULO = −195.0

DELTA = 3.25

EPSIL = .0000001

Table II

Data for Fourth Order Problem

$$\frac{e_o}{e_{in}} = \frac{s^3 + 1.25s + .375}{s(s^3 + 1.877s^2 + 3.81s + 2.0)}$$

$$\underline{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2.0 & -3.82 & -1.87 \end{bmatrix}$$

$$\underline{g} = \begin{bmatrix} 0 \\ 1 \\ -.627 \\ 2.260 \end{bmatrix}$$

TAU = .05

UHI = 7.1

ULO = -7.1

EPSIL = .0000001

DELTA = .02

$$\underline{x}(0) = \begin{bmatrix} 1. \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

when applied to eliminate right hand side derivatives. For the sixth order transfer function the coefficients of the denominator were quite large which led to a rapid buildup in magnitude of the elements of $\underline{g}$. Note that the $\underline{g}$ vector for the fourth order problem does not exhibit this characteristic. Examination of the $\underline{H}$ matrix for the sixth order problem (Table DII, Appendix D) shows that the large values of the last two elements of $\underline{g}$ lead to large values for part of the last two columns of $\underline{H}'$. Experience with computation shows that these large variations in the elements of $\underline{H}$ cause a rather slow descent. The reason for this can be seen from the formula

$$\eta = -\frac{\underline{g}'\underline{g}}{\underline{g}'\underline{H}'\,\underline{R}\,\underline{H}\,\underline{g}} \tag{141}$$

The large values of $\underline{H}$ relative to $\underline{g}$ cause η to become quite small and thus the change in control $\delta \underline{u}$ becomes very small. This last fact also holds for other calculations of η.

Results of the Computation

In this section the results of the computation will be discussed. All of the figures referred to in this section are located at the end of this chapter.

Figure 12 is a plot of the optimum driving function for the fourth order approximation with all four state variables directly controlled. Figure 13 presents a plot of the state variable trajectories which result when the control of Figure 12 is applied. A plot of the terminal norm V for the different control intervals used

is presented in Figure 14. The optimum control interval for a terminal norm V of .082 is 1.85 seconds. Three results should be pointed out. First, the control is essentially "bang-bang". Secondly, the state variables are not all derivatives of one another. This can be immediately explained by noting that

$$x_i \neq \frac{dx_{i-1}}{dt} \tag{152}$$

but that

$$x_i = \frac{dx_{i-1}}{dt} - g_{i-1}\,u \tag{146}$$

Thirdly, note that the negative excursion of x_1 is slightly greater in magnitude than the original error.

A solution for the sixth order system with weighting constants of 6, 5, 4, 3, 2, and 1 was not completed. The descent was attempted but after 30,000 iterations data showed that convergence was going to be quite slow. There is some question as to whether control is possible at all for this case. The reasons for this will become more clear after the data from two of the other problems is reviewed.

The data from the solution for direct control of the first two state variables for both the fourth order and the sixth order systems is to be considered next. The various plots of the optimum control, of the state variables, and of the terminal norm are shown in Figures 15 through 24. Note the differences in scale which were necessary for the plots of the sixth order problem. The control interval

for the V of the fourth order approximation to reach a value of .16 is .65 seconds. In .675 seconds the terminal norm of the sixth order system reached a value of .13. The relay switching occurs for both systems at about the same time. Note also the close resemblance of the trajectories of x_1 and x_2 for both systems. These resemblances suggest that the fourth order system was a good approximation of the sixth order system. Of special interest, however, are the trajectoies of the higher order state variables. Remember that direct control was not attempted for the state variables x_3 through x_6. The plots of the state variable trajectories for the sixth order system illustrate quite well what has happened. The State Space equations for the sixth order system show that

$$x_3 = \frac{dx_2}{dt} \tag{153}$$

but more especially that

$$x_4 = \frac{dx_3}{dt} - u \tag{154}$$

$$x_5 = \frac{dx_4}{dt} + 2.267u \tag{155}$$

$$x_6 = \frac{dx_5}{dt} + 1269.0u \tag{156}$$

The trajectories of the state variables show these relationships graphically. Note especially how the trajectories of x_4, x_5, and x_6 are governed by u. It is now quite apparent why direct control of all the state variables is so difficult to obtain in this problem.

The control is fighting itself in this case. It should be pointed out again that the presence and size of the g_i s is a result of the numerator dynamics of the entire system and the high natural frequency of the servo and actuator.

The trajectories of x_3 and x_4 for the fourth order example cannot be compared with those of x_3 and x_4 for the sixth order system, again because of the different way in which the numerator dynamics influence the solution.

A solution for direct control of the first four state variables of the sixth order system was also attempted. Convergence of the solution was quite slow and therefore a complete solution was not obtained. However, it appears that convergence would be possible if enough iterations were tried.

A term which seems applicable to a problem of this nature is "controllability". A definition of "controllability" is not easily stated, but it is possible to think of it in terms of the number of iterations in the descent necessary to compute the optimum control. The more iterations necessary the less controllable is the system. Of course it is immediately evident that the controllability depends on which of the state variables one wishes to control directly and to what degree. For example, controlling directly all six state variables of the sixth order system to the same degree would be almost impossible. Conversely, the descent for direct control of the first two state variables of the sixth order system was completed

quite rapidly. See Tables D-III through D-V in Appendix D for a comparison of the number of iterations necessary for the different problems.

The results of this section have demonstrated the use of the State Space techniques and the Steepest Descent method in solving a practical problem of computing an optimum driving function. An attempt was made to demonstrate the limitations of the methods as well as the capabilities. Data which can be correlated with the analogue computer simulation of the problem was presented. Attempts at five solutions to the problem were made; three of these were successful.

The next part of this report will be concerned with the analogue computer simulation of the Minneapolis-Honeywell relay servo loop.

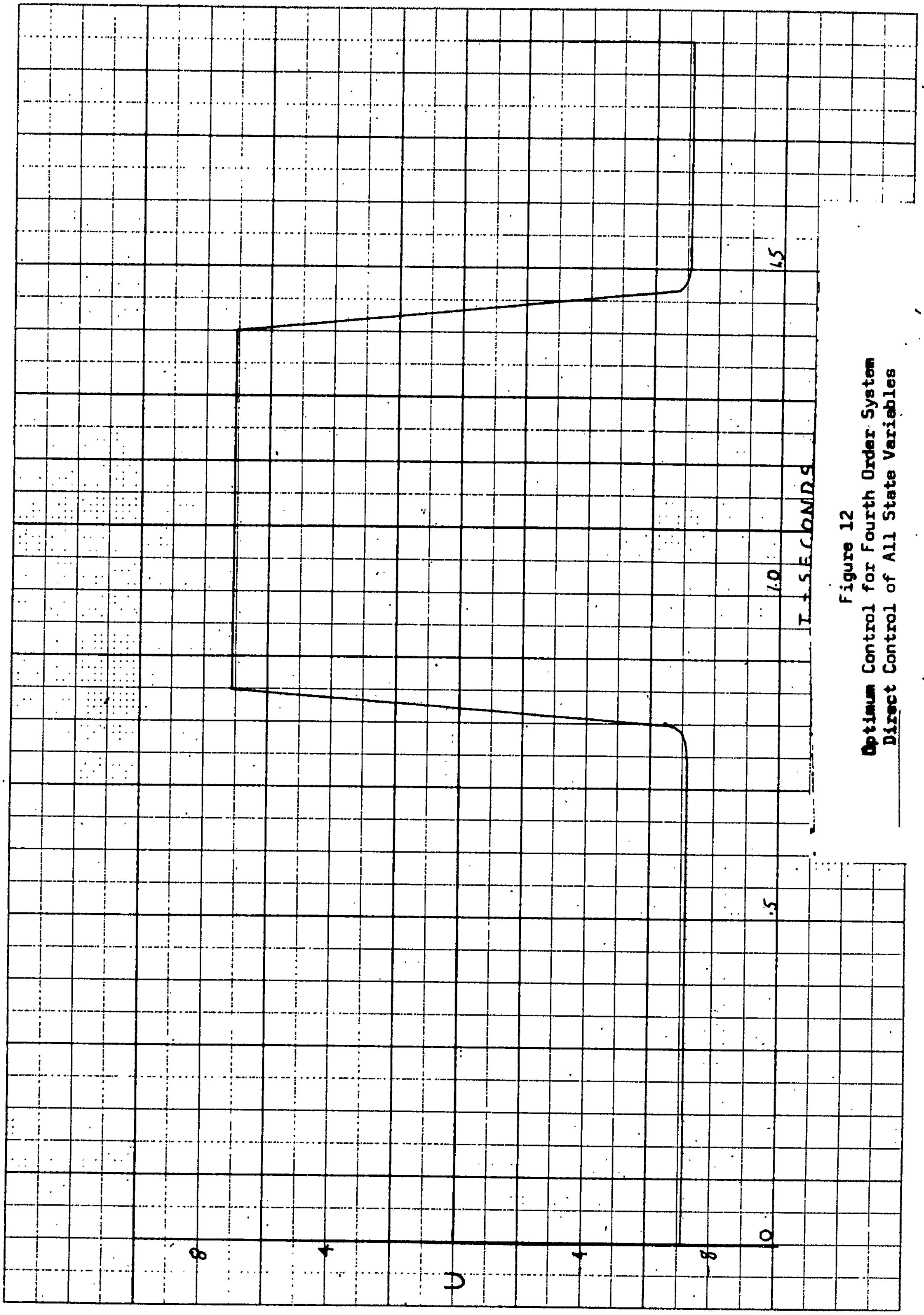

Figure 12
Optimum Control for Fourth Order System
Direct Control of All State Variables

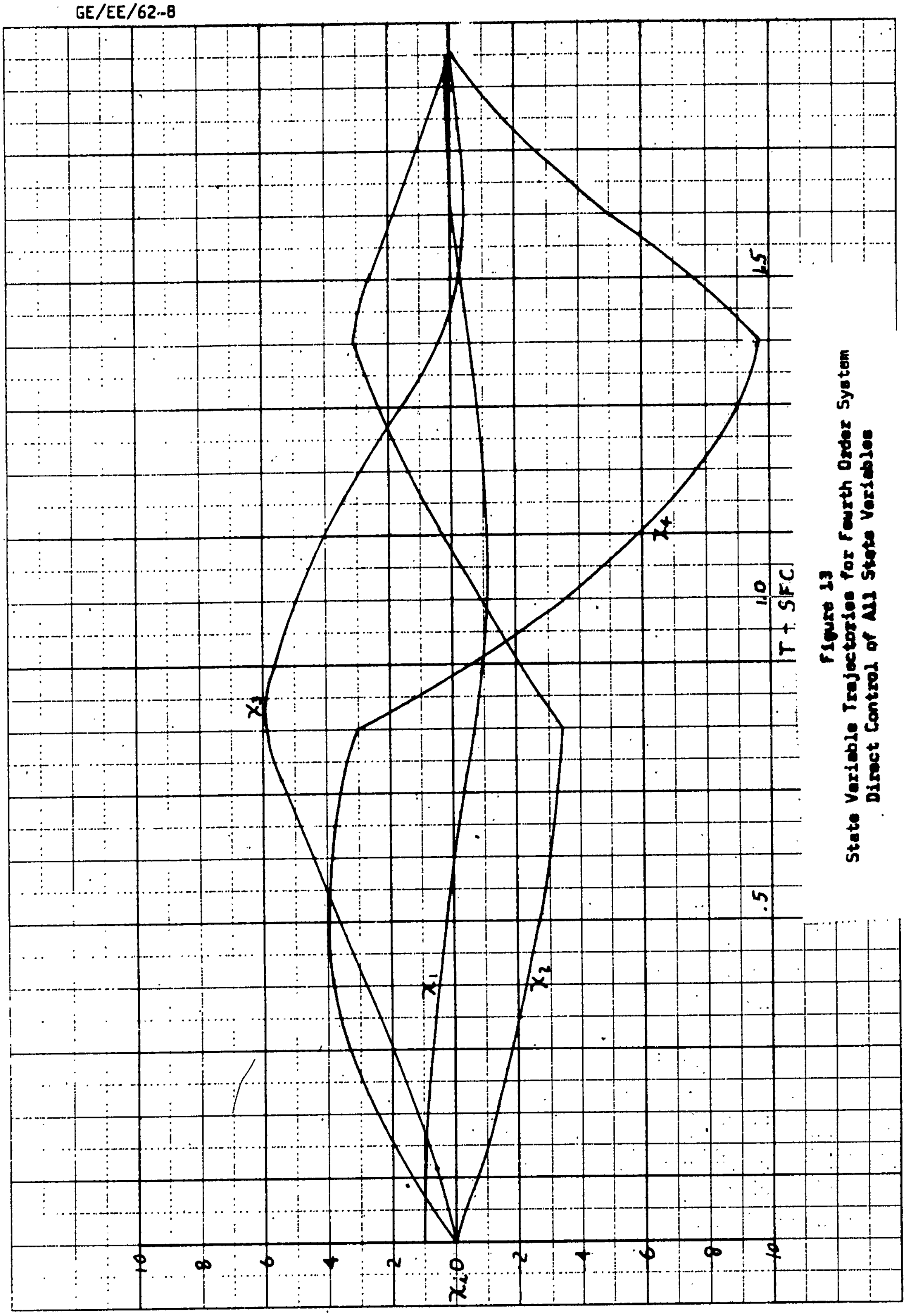

Figure 13
State Variable Trajectories for Fourth Order System
Direct Control of All State Variables

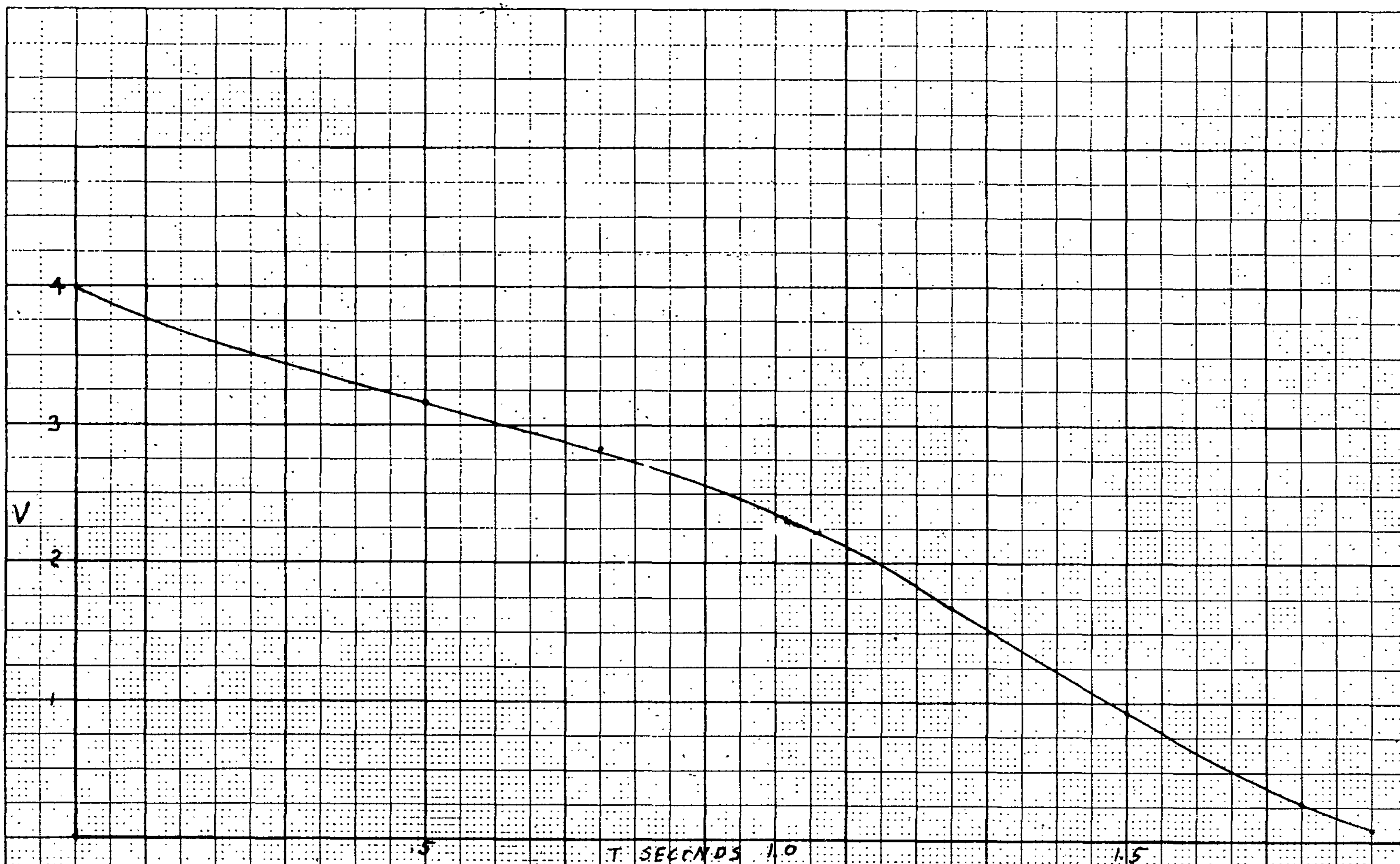

Figure 14
Terminal Norm vs. Control Interval
Fourth Order System
Direct Control for All State Variables

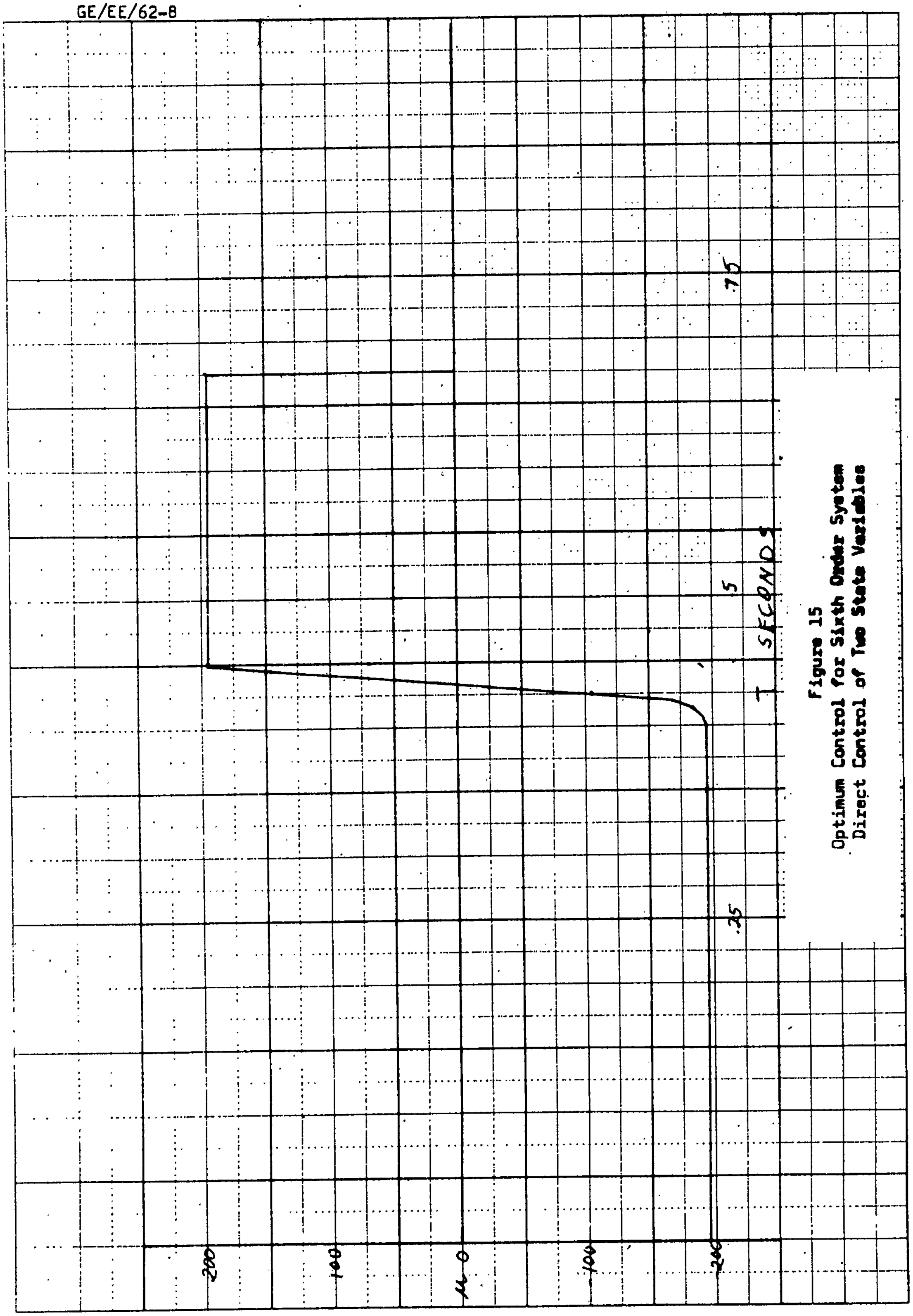

Figure 15
Optimum Control for Sixth Order System
Direct Control of Two State Variables

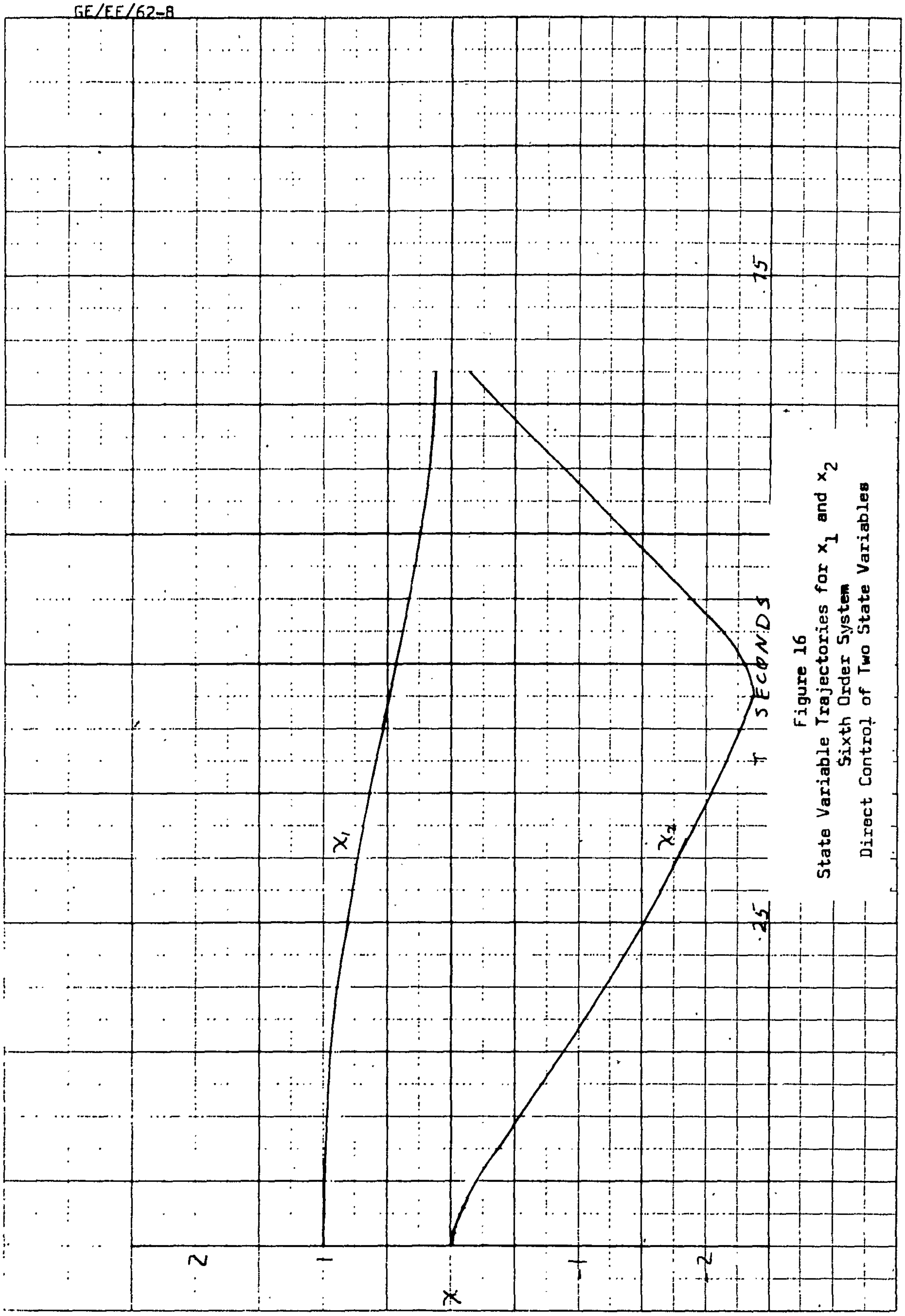

Figure 16
State Variable Trajectories for x_1 and x_2
Sixth Order System
Direct Control of Two State Variables

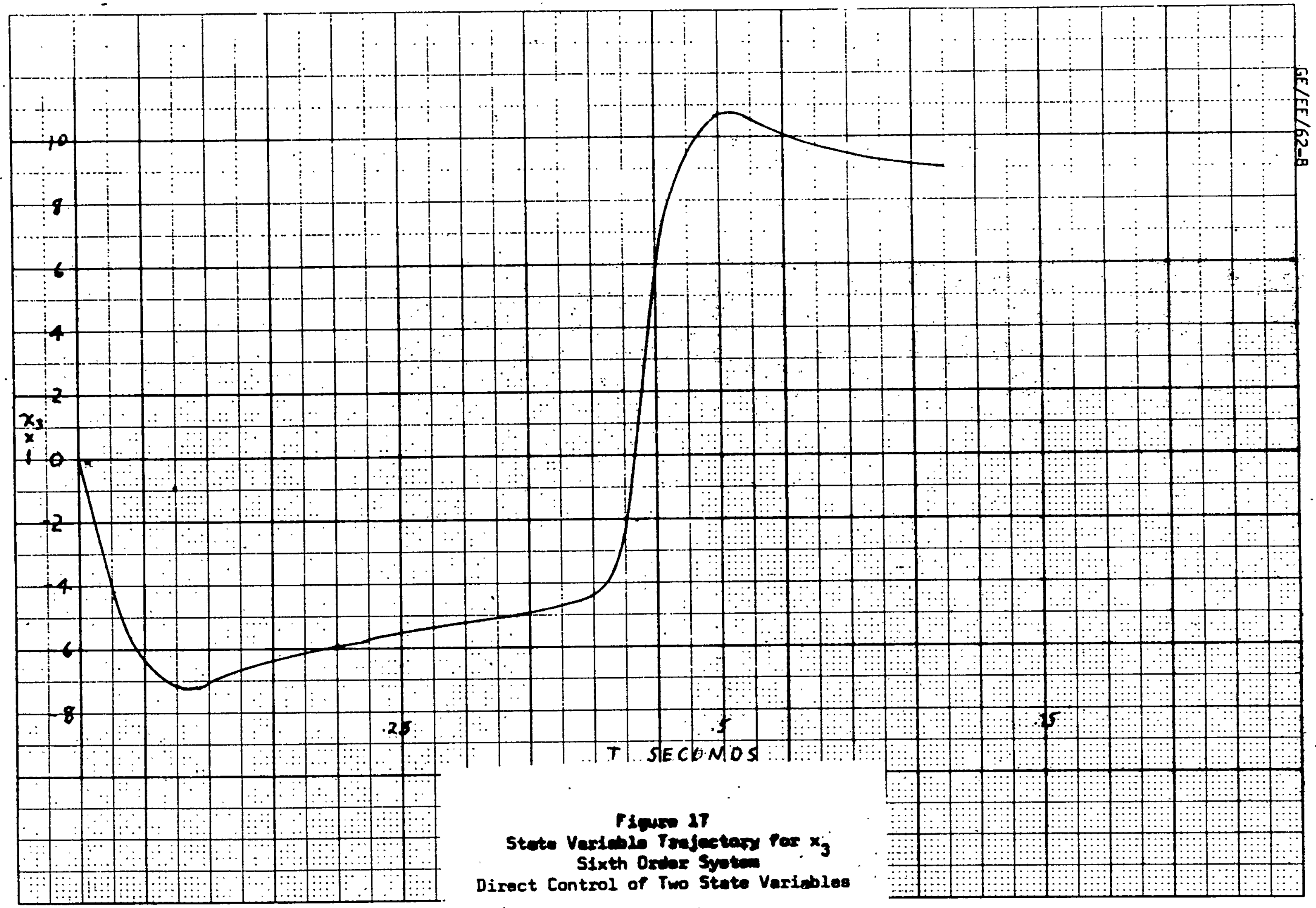

Figure 17
State Variable Trajectory for x_3
Sixth Order System
Direct Control of Two State Variables

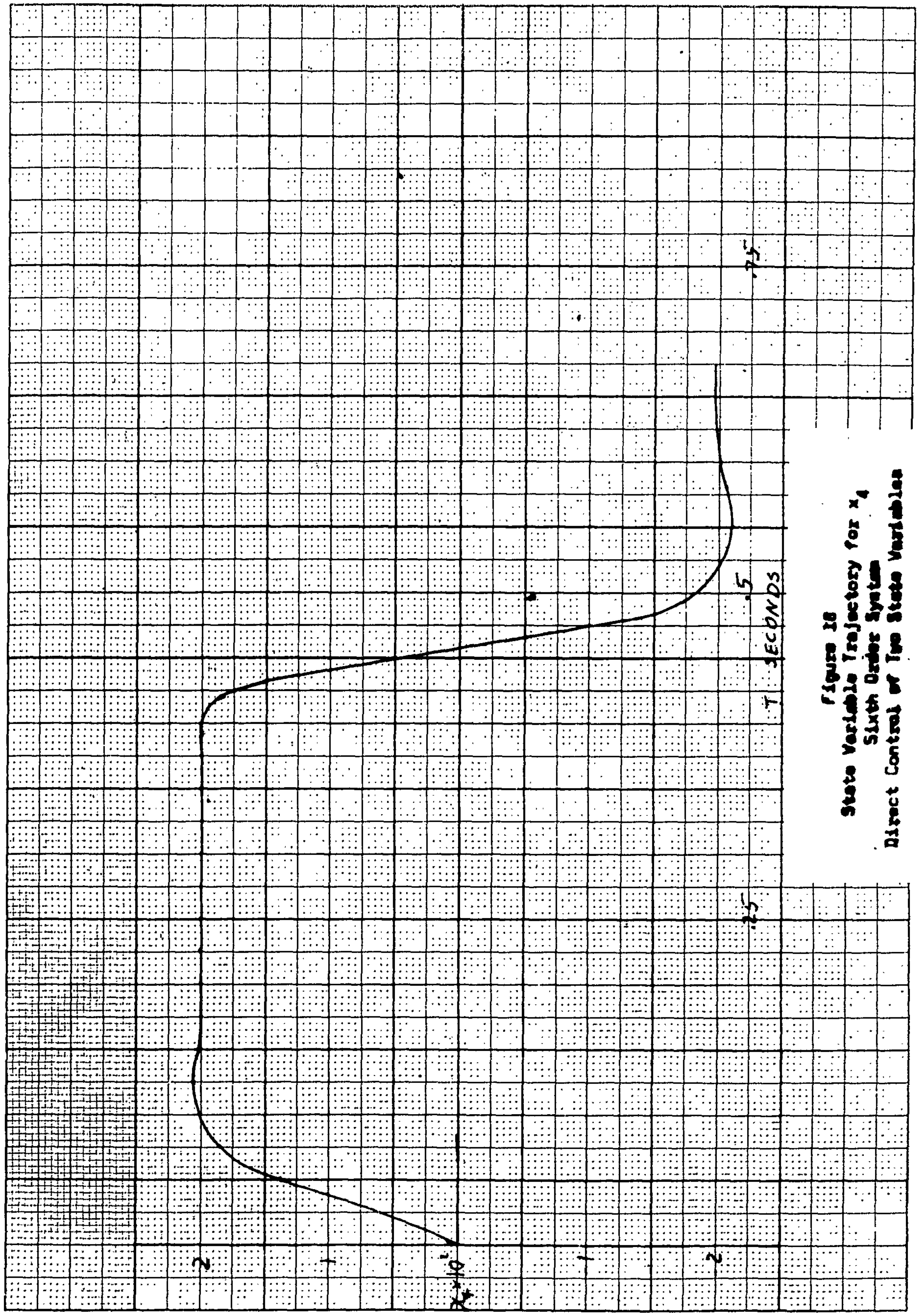

Figure 18
State Variable Trajectory for x_4
Sixth Order System
Direct Control of Two State Variables

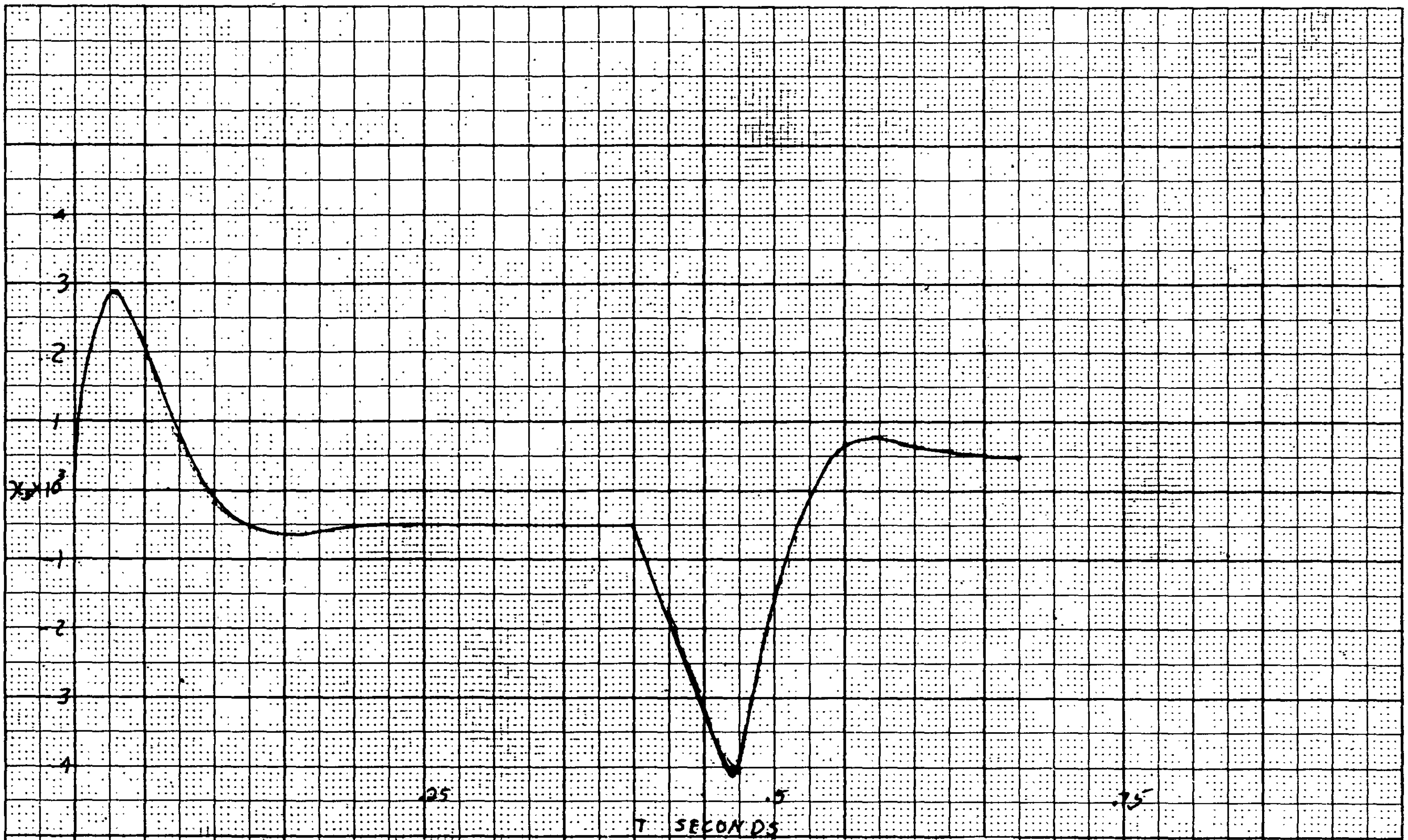

Figure 19
State Variable Trajectory for x_5
Sixth Order System
Direct Control of Two State Variables

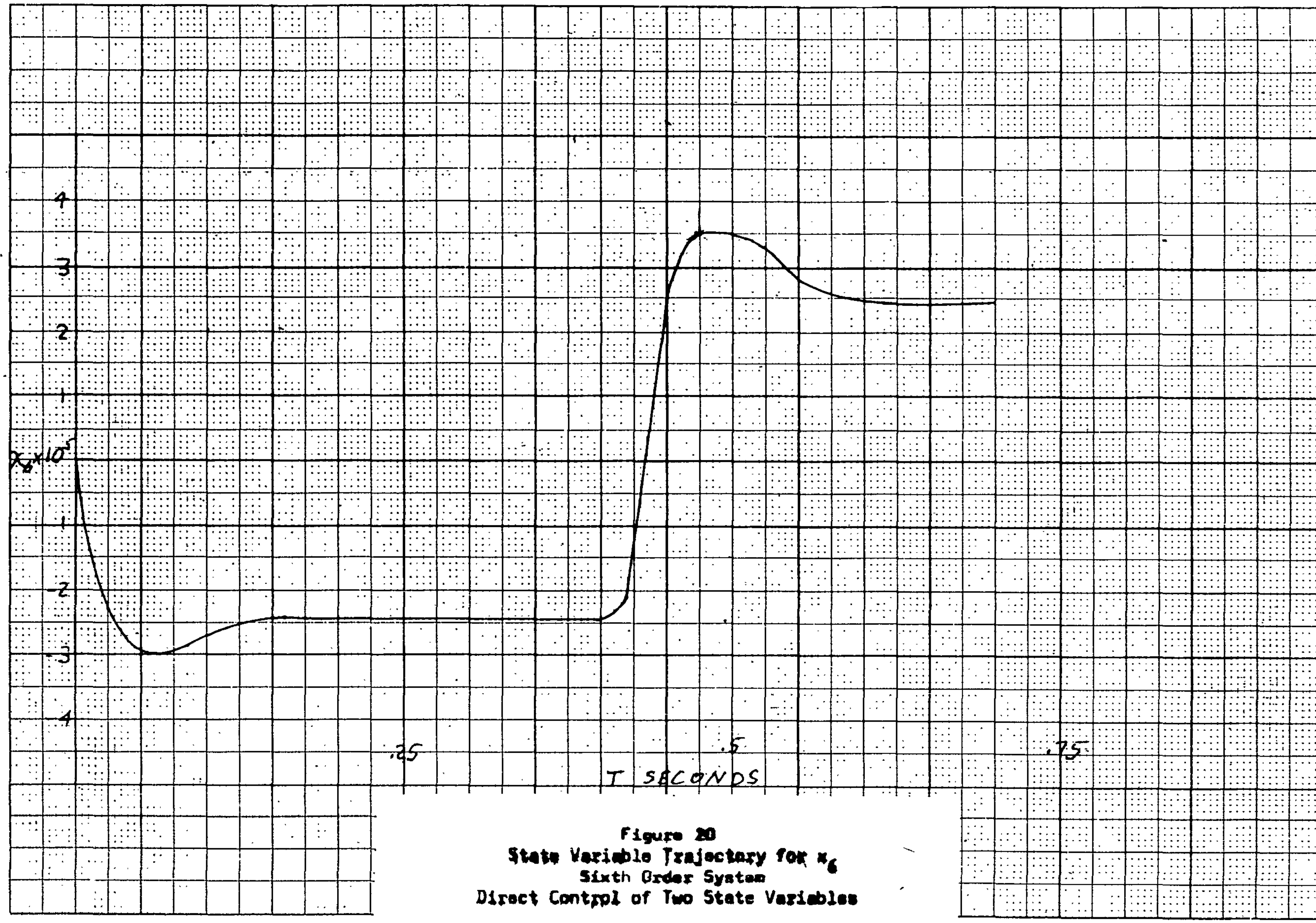

Figure 20
State Variable Trajectory for x_6
Sixth Order System
Direct Control of Two State Variables

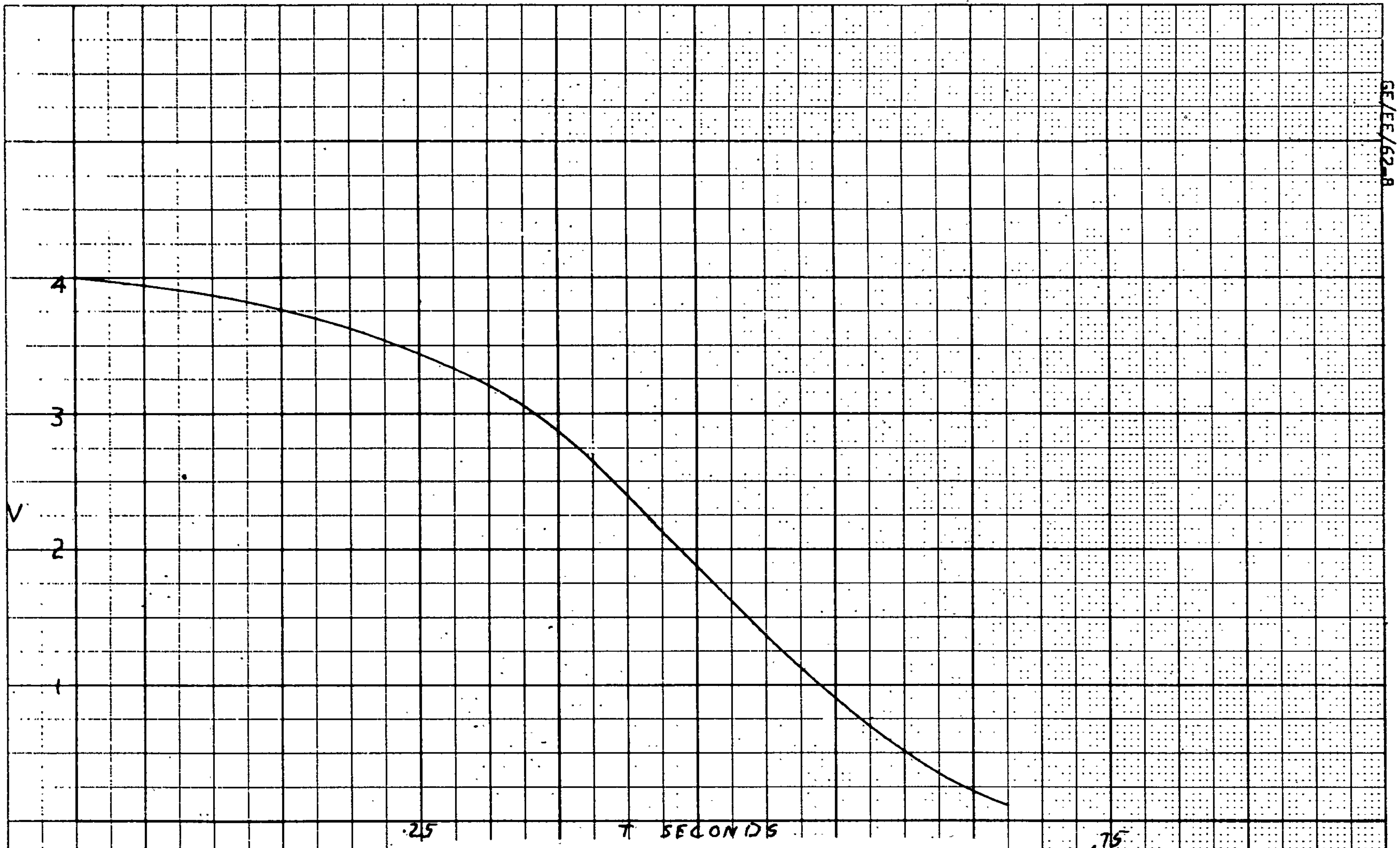

Figure 21
Terminal Norm vs. Control Interval
Sixth Order System
Direct Control of Two State Variables

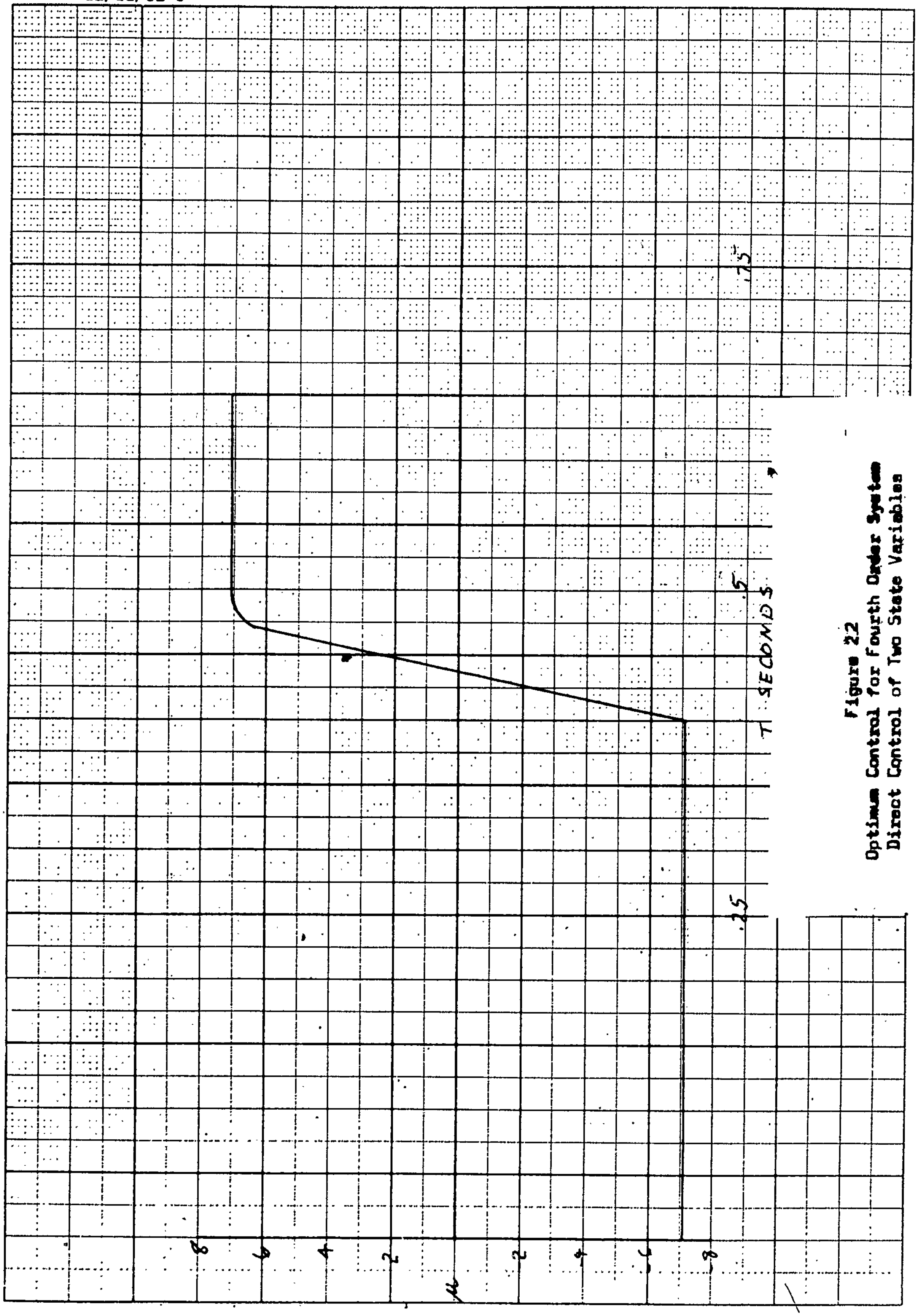

Figure 22
Optimum Control for Fourth Order System
Direct Control of Two State Variables

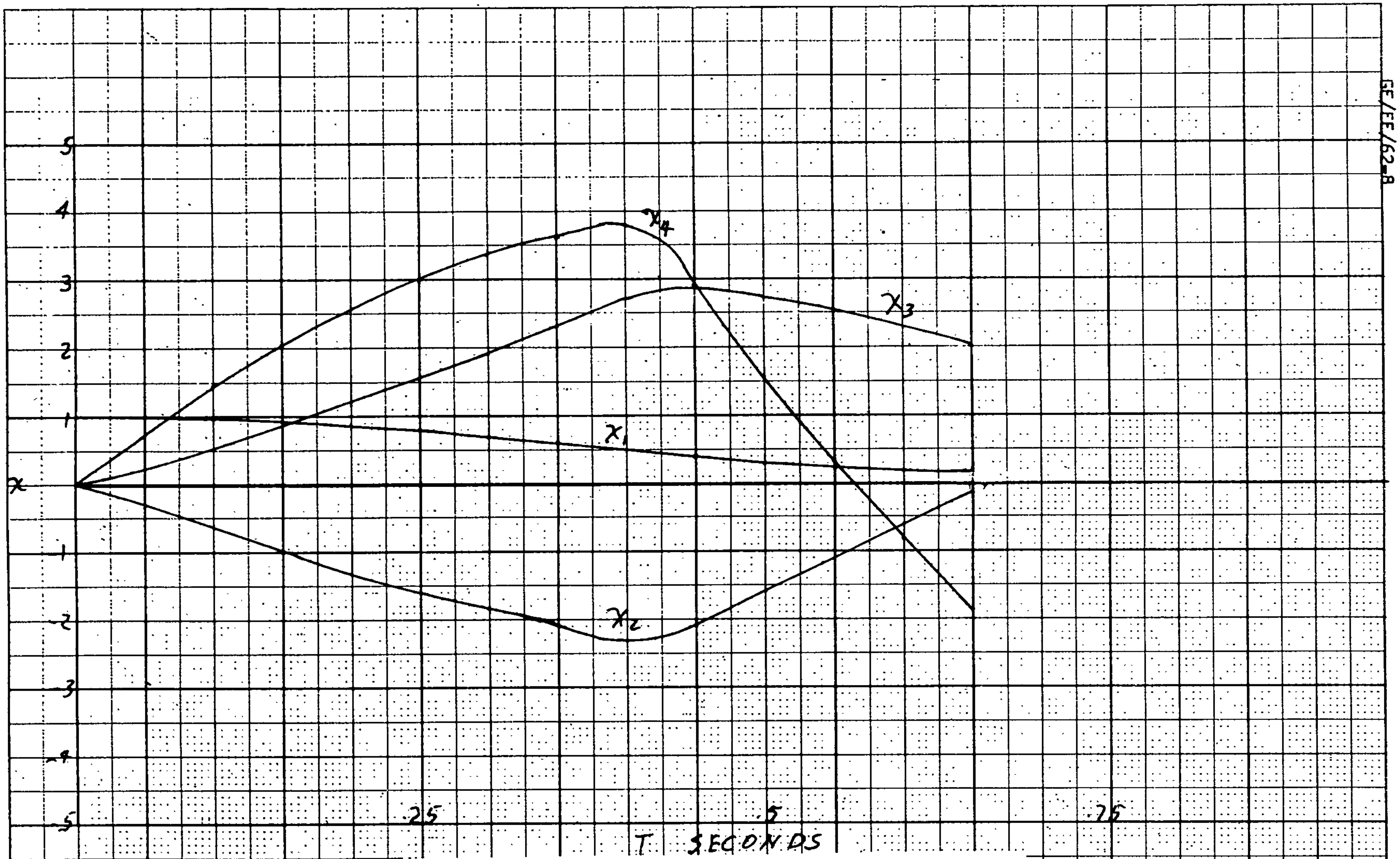

Figure 23
State Variable Trajectories for Fourth Order System
Direct Control of Two State Variables

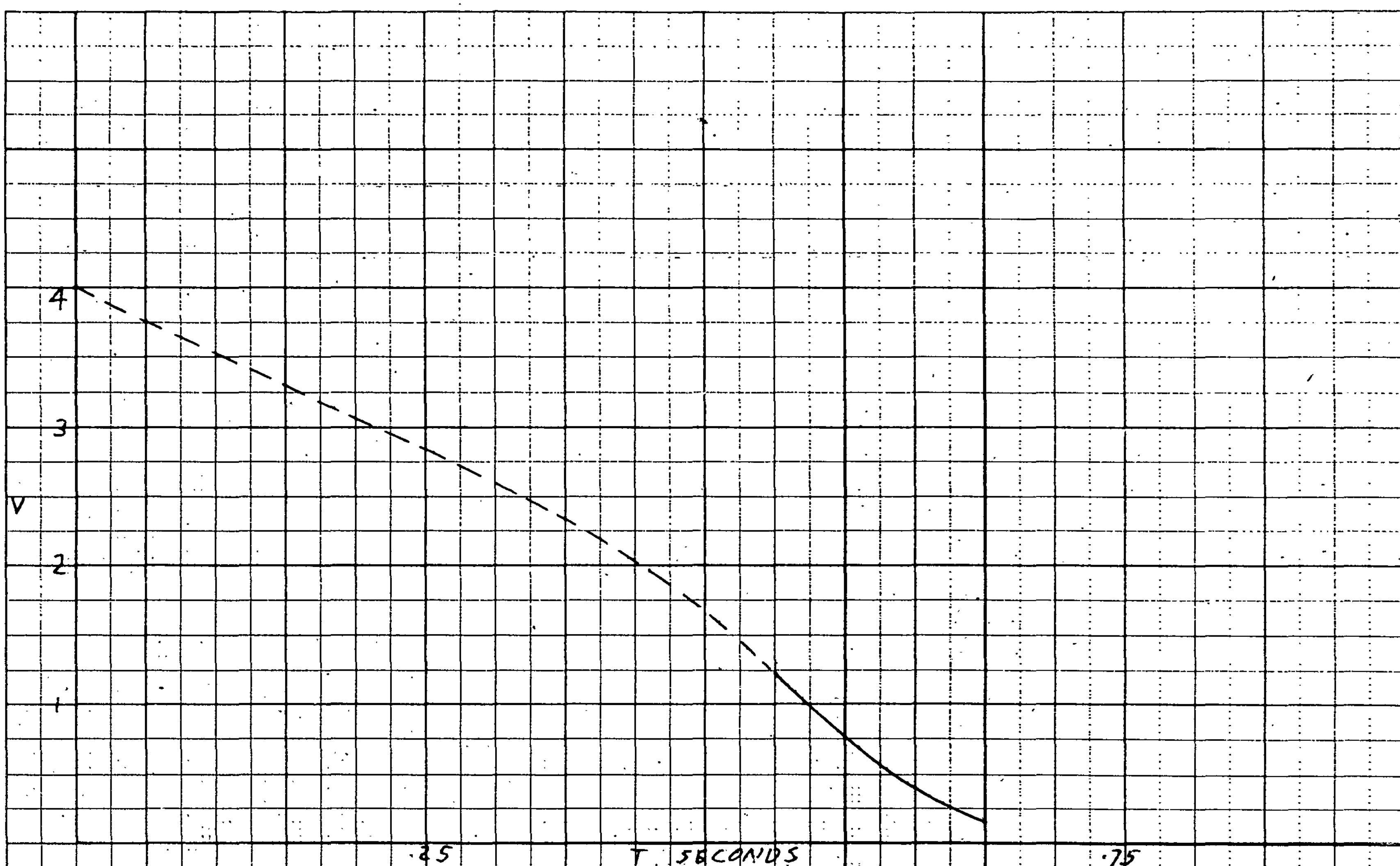

Figure 24
Terminal Norm vs. Control Interval
Fourth Order System
Direct Control of Two State Variables

VI. Analogue Computer Simulation of the Minneapolis-Honeywell Relay Servo Loop

In order to form a basis for comparison of the results of Chapter V an analogue computer simulation of the Minneapolis-Honeywell relay servo loop was performed. Figure 9, which is a block diagram of the relay servo loop is presented at the end of this chapter for the reader's convenience.

In keeping with the analysis of Chapter V, the model was eliminated from the original Honeywell Adaptive Control System, as were the gain changer and limiter and the AC dither. The response of the relay servo loop with a step input applied is that which is desired. The relay output and the system output can then be compared with optimum driving function and the state variable trajectories obtained by the calculations of Chapter V.

Description of the Simulation

The simulation of the relay servo loop is quite straightforeward and will not be discussed in detail.The procedures outlined in Chapter 19 of Automatic Control Systems Analysis and Synthesis were followed.(Ref 2:453-475).

The following points concerning the simulation should be noted however: 1) due to the high natural frequencies of the servo and acutator and the rate gyro it was necessary to time scale the problem by slowing the solution by a factor of ten; 2)the electronic relay

used in the Honeywell loop was replaced by a high gain amplifier in cascade with a differential relay. Figure 26 at the end of this chapter is a diagram of the simulation.

A describing function analysis assuming an ideal relay with an output of 1.5 volts was used to determine stability and the approximate frequency of the limit cycle to be expected. A Log Magnitude Phase Angle Diagram for the describing function analysis is shown in Figure 27 at the end of the chapter.

Results of the Simulation

Step inputs of one volt were applied to the system after a steady state limit cycle had been established. The one volt corresponds to the one unit of initial error used in the computation of the optimum driving function in Chapter V. Figure 25 at the end of this chapter is a typical analogue computer recording for the response to the unit step input. The traces are, beginning at the top, error signal, input to the high gain amplifier, relay output, and system output, in that order. Upon application of the step input, initial switching of the relay took place, if the polarity wasn't already such as to drive the error to zero. As the relay input crossed the aero axis the relay switched polarity. Finally, when the error approached zero limit cycle operation was resumed.

Results of high speed recorder runs show that the error is reduced to zero in about .75 seconds. The limit cycle is resumed

after about .65 seconds. The average time from the detection of the initial error until relay polarity reversal is bout .4 seconds. Exact times are difficult to determine because they depend upon the state of the system in the limit cycle to some extent.

It should again be pointed out that the input to the high gain amplifier is the error signal plus .225 of its derivative. The trace of the error signal shows that for a negative error the slope is positive; thus, when the error decreases below .225 of its derivative at that point, the relay reverses polarity. This reversal prevents overshoot. Finally, note that the shape of the trace of the system error resembles the shape of the trajectory of x_1 in Figure 16 which is located in the preceding chapter.

The data just presented leaves little more to be said concerning the analogue computer simulation. The reader should, however, keep the results in mind since they will be compared in the next chapter of this report with the results of the computations for optimum control.

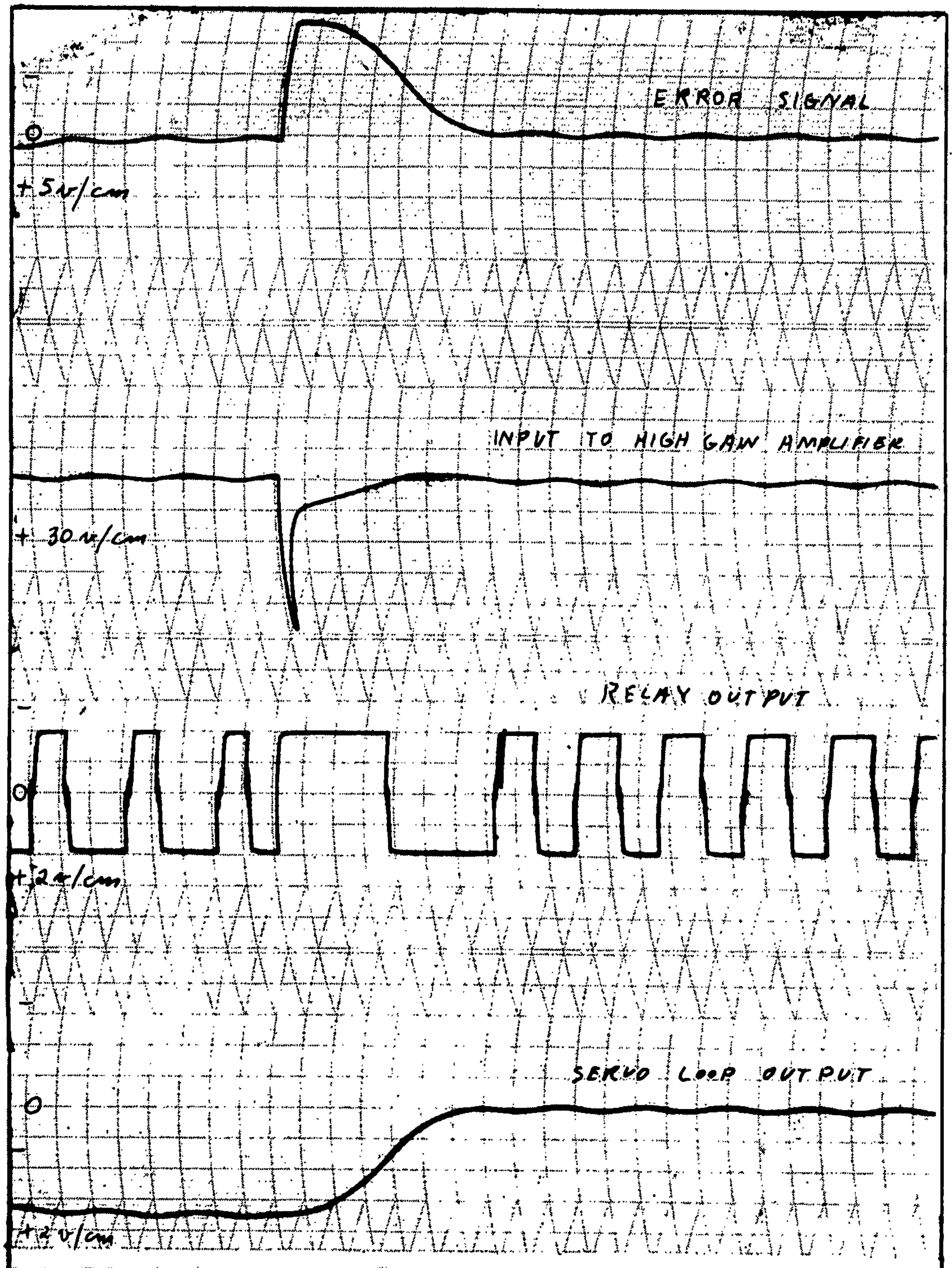

Figure 28
Results of Analogue Computer Simulation

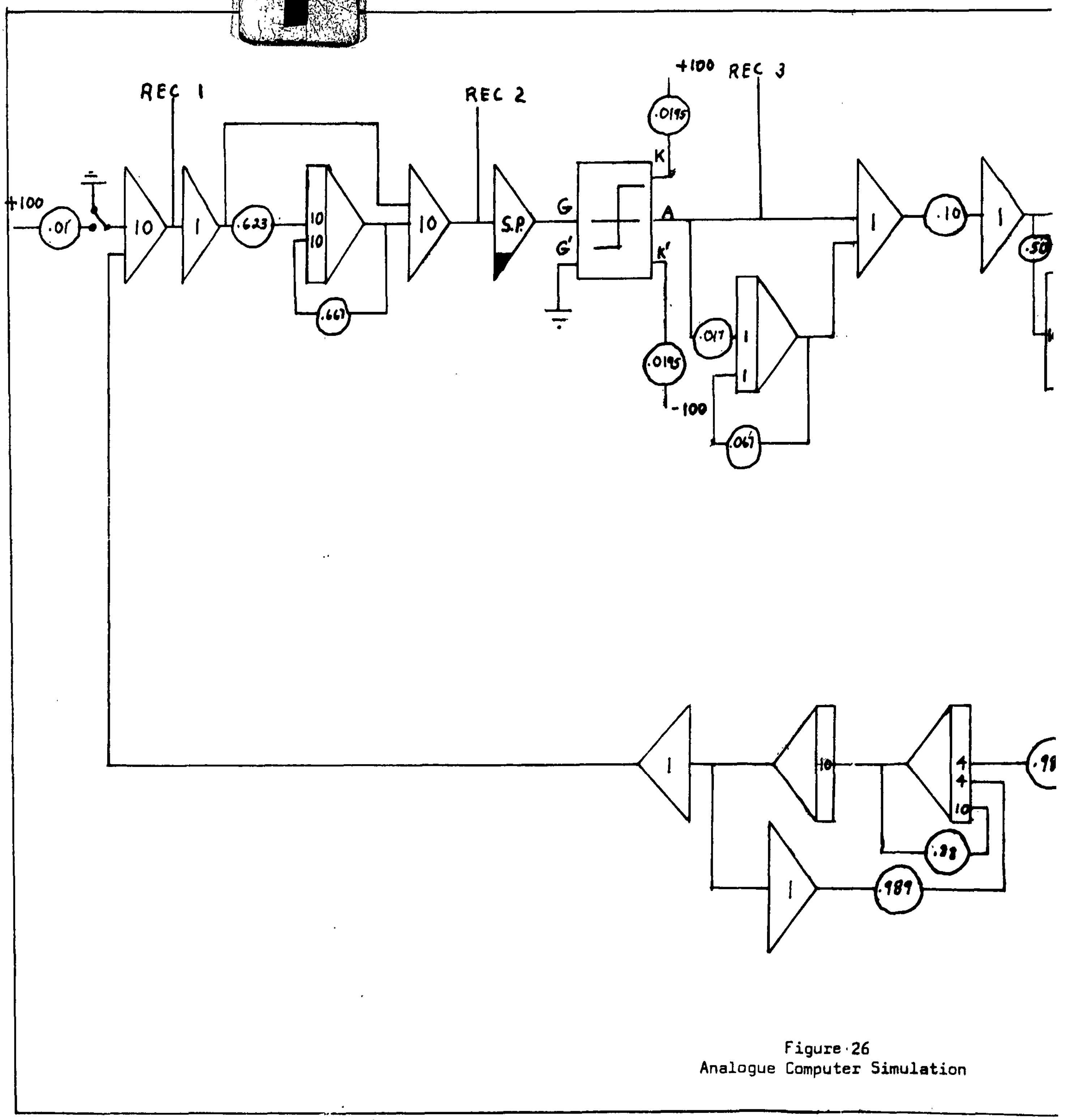

Figure 26
Analogue Computer Simulation

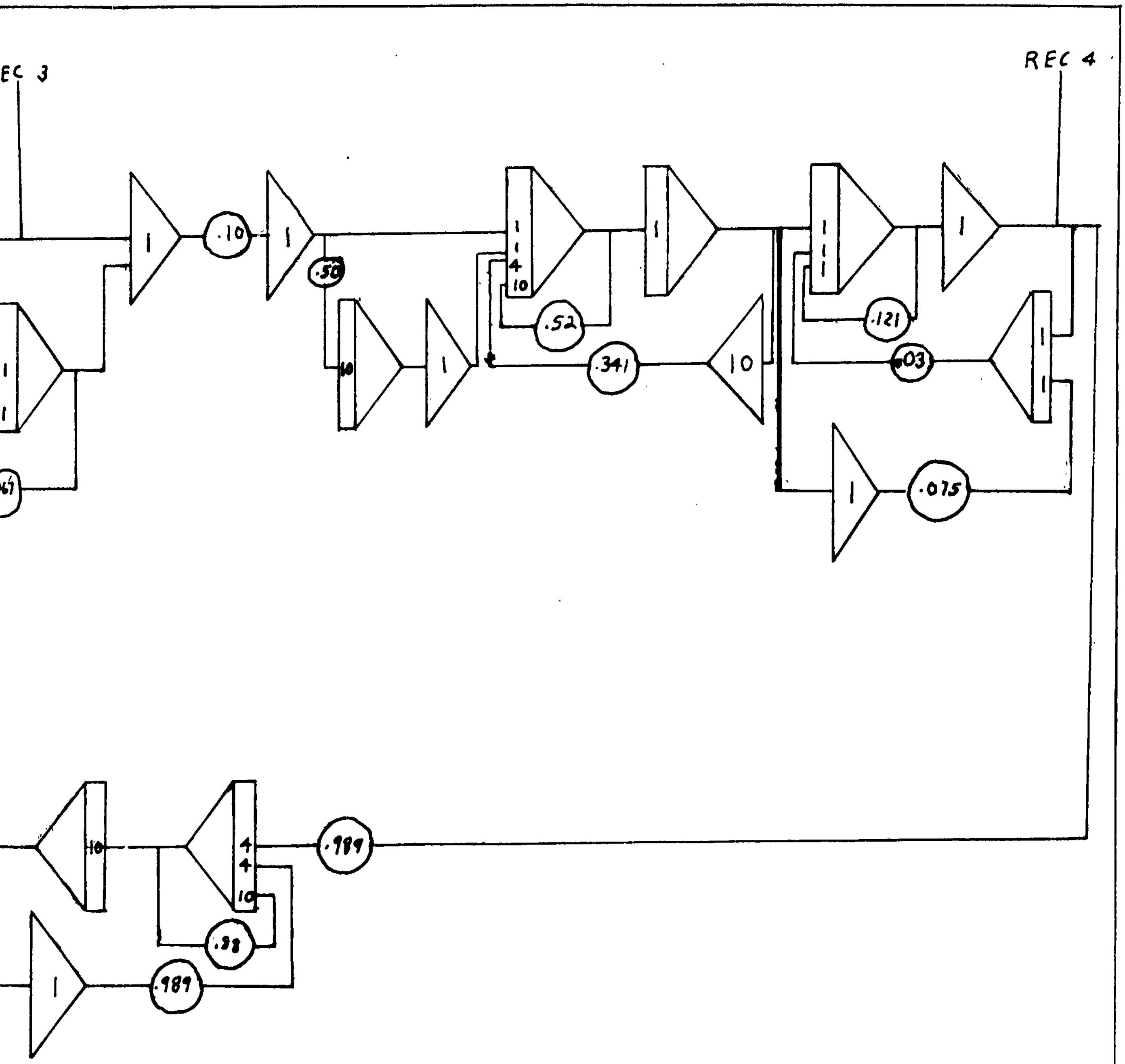

Figure 26
ogue Computer Simulation

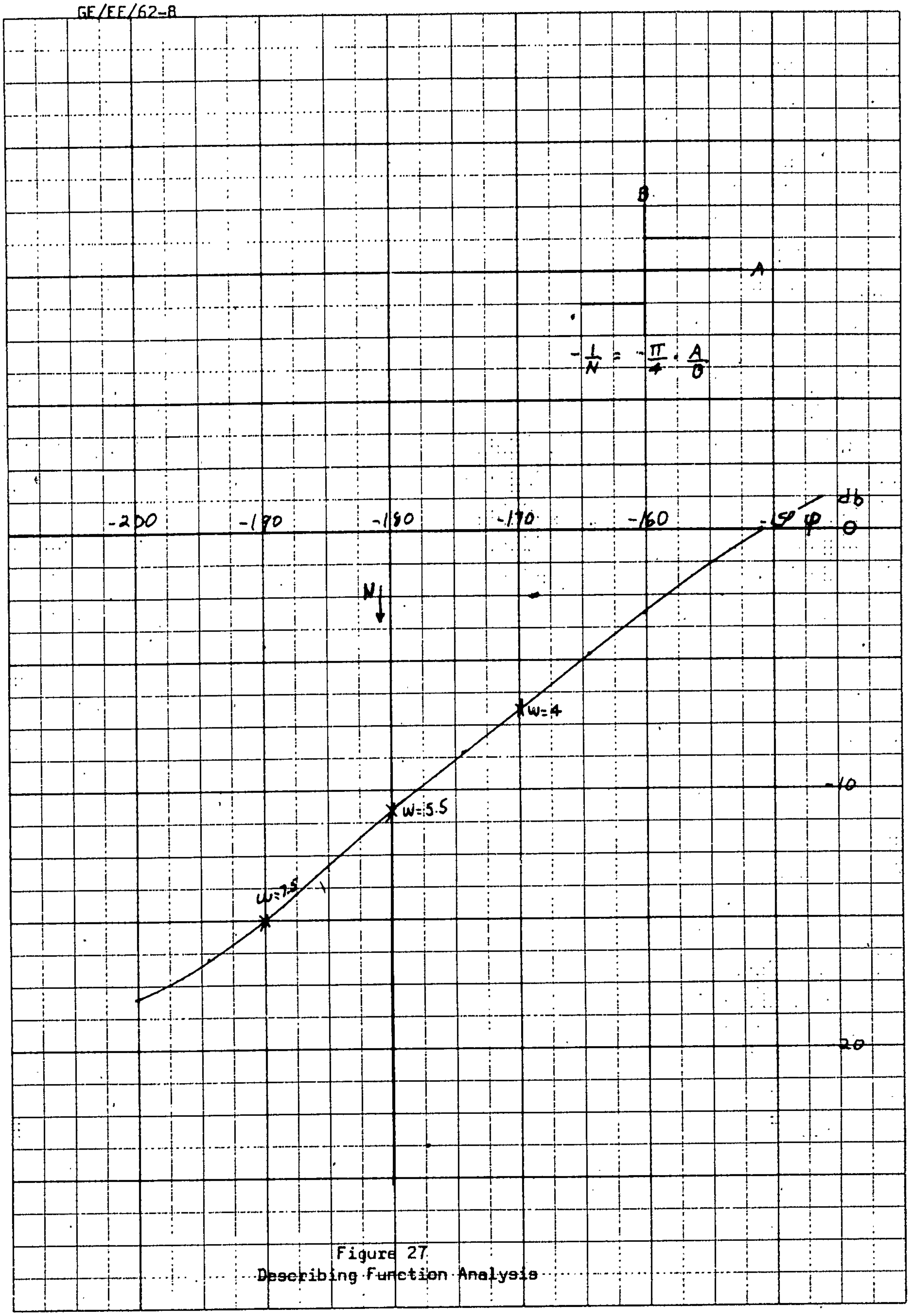

Figure 27
Describing Function Analysis

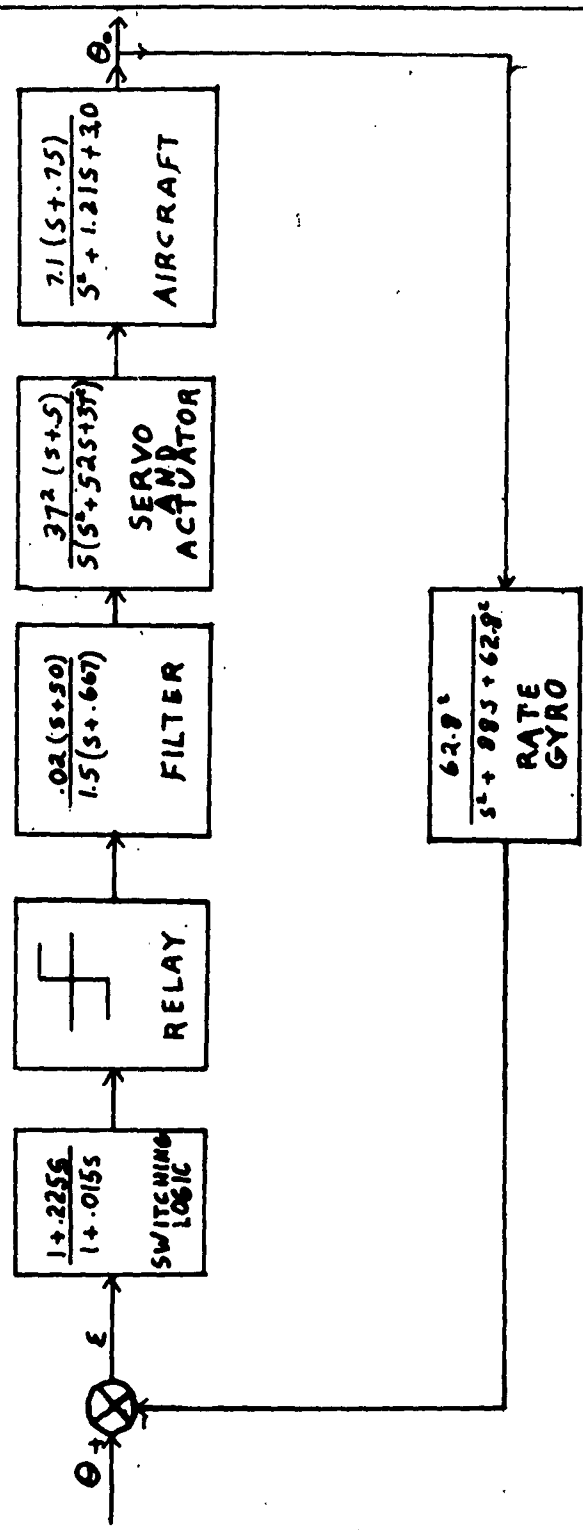

Figure 9
Relay Servo Loop Used in Analysis

VII. Conclusions and Recommendations

In this last chapter of this report the data compiled in the previous sections will be summarized and conclusions will be drawn from it. Recommendations for further study of certain phases of this report will then be made.

It is not the purpose of this chapter, or of this report, to attempt to determine the quality of the Minneapolis-Honeywell Adaptive Control System just discussed. Rather it is to demonstrate the application of State Space and optimization techniques by analyzing a practical problem.

Summary of Data

In this section data from the previous chapters will be summarized. Also, from time to time, the figures of Chapters V and VI will be referred to without stating their location.

It is possible to compare only part of the data from the computations for the optimum driving functions to that from the analogue computer simulation. The data that may be compared are the results of the optimum solution to the fourth and sixth order problems for direct control of the first two state variables. A glance at Figures 15 and 22 shows that the driving function calculated compares very well with the output of the relay as recorded during analogue simulation. The first relay polarity reversal for both computational solutions

occured at about .4 seconds, as did the first polarity reversal during analogue simulation. The simulation also showed the error driven to essentially zero in about .7 seconds. This corresponds quite well to a terminal norm of .13 with a control interval of .675 seconds. Of course it is impossible to state that a direct correlation exists because the criterion formed by the switching logic of the Honeywell relay servo loop and that formed by the terminal norm function are different. However, the simularity of the two results does indicate that for this particular problem the criterion function used for the computation of the optimum driving function is meaningful.

Note the shape of the trajectory of x_1 which is plotted in Figure 16. It is about the same as that of the trace of the error signal which was recorded during the analogue simulation.

Another point of interest in the results of the digital computation is the shapes of the trajectories of the various state variables. Comparison of Figure 13 with Figure 23 shows that when an attempt was made to directly control x_3 and x_4 in the fourth order approximation their excursions were greater during the control interval than when they were allowed to follow without direct control. Of course x_3 and x_4 were not driven to zero in the latter case.

Another point of interest is that a longer control interval was necessary to control directly four state variables than to control two; thus, the expenditure of energy by the system had to

be greater for the longer control interval. Also, one sees from Figure 13 that the maximum negative excursion of x_1 was greater than the initial error in the system.

A brief glance at the tabular data in Appendix D shows the number of iterations necessary to attain convergence varies over a wide range for the different problems considered. Generally, but not in all cases, the number of iterations necessary increases as the optimum control interval is approached. Also, the more difficult it is to control certain state variables, the greater is the number of iterations necessary for convergence. In general, then, the rate of convergence is a measure of the "controllability" of the system. The rate of convergence may be measured by the number of iterations necessary to attain convergence.

Conclusions

On the basis of the data presented in the preceding section and that presented in discussions elsewhere in this report several interesting conclusions may be drawn. Probably the most important and far reaching of these is that each solution for the optimum driving function is in reality a sub-optimization. The results of Chapter V showed how much the optimum control interval and the trajectories of the state variables depended on the matrix $\underline{R}$. It goes without saying that a different function for the terminal norm would also have influenced the results a great deal. It should be understood, however, that the computed driving function is optimum

for the terminal norm chosen. The question remains of whether the terminal norm used is the best criterion. Thus the problem of picking a criterion to minimize still remains a matter of judgement. For example, consider the particular system analyzed in this paper. Compare the solution for the fourth order system with the diagonal of $\underset{\sim}{R}$ equal 5, 1.12, 0, 0, and that for the same system with the diagonal of $\underset{\sim}{R}$ equal 4, 3, 2, 1. If one is interested in reducing the error, which in this case is the first state variable, to zero, it appears better to attempt to control directly only the error and its derivative than to attempt to control directly all the state variables. Certainly the optimum control interval is less for the former.

It should again be emphasized that the criterion function chosen is reasonable, although some of the values used for the $\underset{\sim}{R}$ matrix might have been a bit unrealistic. A further point along these lines is that attempts to control state variables which are dominated by [illegible] will probably meet with very slow convergence or even lack of convergence.

The question of "controllability" and the related computation time, is, for this report, somewhat academic, except in cases where it is impossible to obtain convergence without many thousands of iterations. However, one problem for which computation time would be important is that of an adaptive state vector control system. Basically, a system of this type would operate in the following way. The state

variables would continuously be measured. Consider that they represent the error of the system. The necessary input to a linear controller to drive these variables to zero would then be continuously computed in real time and applied to the controller. For a linear controller with good controllability such computation in real time seems feasible by the Steepest Descent method provided a very high speed computer is employed.

Finally, the fact that all the solutions for the optimum driving function for the examples investigated in this report were "bang-bang" is of interest. The same results were obtained whether the Gradient Projection Descent Scheme or the Corner Aiming Descent Scheme was employed. This result is in direct support of the maximum principal of Pontriagin; in addition, it should be pointed out that the "bang-bang" solutions were arrived at by an entirely different mathematical approach than was the proof of the maximum principal.

Recommendations

Unfortunately, there were many interesting phases of the problem left untouched in this investigation due to the element of time.

Also it should be pointed out at this time that the computer programs used and a brief explanation of each are presented in Appendix A. Hopefully enough information has been provided for the interested reader to utilize the programs with little added study.

One of the phases of the analysis which remains unexplored is further experimentation with the matrix $\underset{\sim}{R}$. Along this same line is the testing of different criterion functions used in the descent.

One criterion function of particular interest would be

$$\operatorname{MIN} \sum_{i=1}^{m} V_i = \sum_{i=1}^{m} r_i \, x_i(T) \tag{157}$$

The computation of the optimum driving function to be applied to a non-linear controller is also possible by the method of Steepest Descent. The interested reader is referred to Part V of Reference 5.

In the field of digital computer programming it would be quite useful to modify the existing descent program to seek out the optimum control interval automatically. Of course due to the computation times involved the use of such a program would have to be confined to systems with a reasonable amount of controllability.

Finally, a study to determine the feasibility of using the Steepest Descent method in an adaptive state vector control system could be investigated.

The above list of possible topics in connection with this report is by no means complete. However, it is hoped that some stimulating ideas were presented.

In this report State Space techniques and the Steepest Descent method were employed to obtain the optimum driving function for a practical linear controller in a relay servo loop. It is felt that the results obtained demonstrate the limitations and capabilities of this type of analysis quite clearly.

Bibliography

1. Coddington, Earl A., and Leninson, Norman. Theory of Ordinary Differential Equations. New York: McGraw-Hill Book Co., 1955.

2. D'Azzo, J. J., and Houpis, C. H. Feedback Control System Analysis and Synthesis. New York: McGraw-Hill Book Co., 1960.

3. Frazer, R. A., Duncan, W. J., and Collar, A. R. Elementary Matrices. New York: The MacMillan Company, 1946.

4. Hildebrand, F. B. Methods of Applied Mathematics. New York: Prentice-Hall, Inc., 1952.

5. Ho, Y. C., and Brentani, P. B. Computational Solution of Optimal Control Problems. Minneapolis-Honeywell Military Products Group Report 1529-TR5. Boston: Minneapolis-Honeywell Regulator Company, March, 1962.

6. Laning, J. H., Jr., and Battin, R. H. Random Processes in Automatic Control. New York: McGraw-Hill Book Co., 1956.

7. Kelley, H. J., Method of Gradients. Chapter 6 of Optimization Techniques edited by George Leitmann. Berkeley: Academic Press, 1961.

8. WADC TR 57-349. A Study to Determine an Automatic Flight Control Configuration to Provide a Stability Augmentation for a High Performance Supersonic Aircraft. Wright-Patterson Air Force Base, Ohio: WADC, May, 1958.

9. WADC TR 59-49. Proceedings of the Self Adaptive Flight Control Systems Symposium. Wright-Patterson Air Force Base, Ohio: ARDC, March, 1959.

10. Yates, R. C. Differential Equations. New York: McGraw-Hill Book Co., 1952.

11. Zadeh, L. A. "From Circuit Theory to System Theory." Proceedings of the IRE, 50:856-865 (May 1962).

Appendix A

Listing and Explanation of the Digital Computer Programs

The purpose of this appendix is to present listings of the digital computer programs used for the computation of the optimum control together with a brief explanation of each program. It is assumed that the reader has a knowledge of the FORTRAN programming system. In writing the programs several changes in notation were necessary. Table A-I on the next page is a list of each important symbol used in the programming together with the corresponding symbol used in Part IV. The reader may wish to refer to this table from time to time in case of confusion.

Calculation of ALIT by Runge-Kutta

In the first program which is listed on pages 104 and 105 $\underline{\alpha}$ is calculated by the method of Runge-Kutta. This method is developed in Part IV of this report. The program is written in 1620 AFIT FORTRAN. The following explanation refers to the listing of the program.

The values for Δt, for the order of the system, and for the control subinterval are first read in as DELT, N, and TAU respectively. The $\underline{F}$ and $\underline{g}$ matrices are then read in by columns. Statements 5 through 14 perform the Runge-Kutta calculation for $\underline{\Delta x}$. A test is then made to see if the control subinterval TAU has been reached. If it has not, control is returned to statement 5 where the calculation is performed again with the new value of $\underline{x}$. When TAU is finally reached $\underline{\alpha}$ is read out and punched on cards.

Table A-I

Meanings for Computer Symbols

Computer Symbol	Symbol in Text
N	n
DELT	Δt
TAU	τ
F	$\underline{F}$
G	$\underline{g}$
DX	$\underline{dx}$
A (Descent Program	$\underline{P}$ or $P(k\tau)$
A (Program for $\underline{H}$)	$\underline{a}$ or $\underline{x}(0)$
JE	k
KA	$K-1$
H	$\underline{H}$
U	$\underline{u}$
UHI	γ_h
ULO	γ_l
EPSIL	ε
DELTA	δ
C	$\underline{G}$
P	$\underline{R}$
XT	$\underline{x}(T)$
W	$\underline{w}$
ETA or ETA2	η, η_1 or η_2
V	V

```
C     CALCULATION OF ALIT BY RUNGE KUTTA
      READ,DELT,N,TAU
      DIMENSION F(6,6),G(6),XNO(6),DX(6,4)
      DO 1 K=1,N
      DO 1 I=1,N
    1 READ,F(I,K)
      DO 2 K=1,N
    2 XNO(K)=0.
      DO 3 K=1,N
    3 READ,G(K)
      P=0.
    5 DO 7 I=1,N
      Z=0.
      DO 4 K=1,N
    4 Z=Z+F(I,K)*XNO(K)*DELT
      K=I
    7 DX(K,1)=Z+G(K)*DELT
      DO 9 I=1,N
      Z=0.
      DO 8 K=1,N
    8 Z=Z+F(I,K)*(XNO(K)+DX(K,1)/2.)*DELT
      K=I
    9 DX(K,2)=Z+G(K)*DELT
      DO 11 I=1,N
      Z=0.
      DO 10 K=1,N
   10 Z=Z+F(I,K)*(XNO(K)+DX(K,2)/2.)*DELT
      K=I
   11 DX(K,3)=Z+G(K)*DELT
      DO 13 I=1,N
      Z=0.
      DO 12 K=1,N
   12 Z=Z+F(I,K)*(XNO(K)+DX(K,3))*DELT
      K=I
   13 DX(K,4)=Z+G(K)*DELT
      DO 14 K=1,N
   14 XNO(K)=XNO(K)+((DX(K,1)+2.*DX(K,2)+2.*DX(K,3)+DX(K,4))/6.)
      P=1.+P
      IF(TAU-P*DELT) 15,15,5
   15 TYPE 16
   16 FORMAT (/18H FOLLOWING IS ALIT)
      DO 17 K=1,N
      PUNCH 45,XNO(K)
   17 TYPE 44,XNO(K)
   44 FORMAT (/E14.8)
```

```
45 FORMAT (2X,E14.8)
   STOP
   END
```

Calculation of Large A and H by Runge-Kutta

The program for the calculation of $\underline{\rho}$ and $\underline{H}$ by the Runge-Kutta method is listed beginning on the next page. There are two forms of the program included, one written in 1620 AFIT FORTRAN and the other written in 7090 FORTRAN. The only differences in the two are the input and output statements and the method of controlling the use of the options. When it is desired to use a large number of control subintervals and the order of the problem is large it is advantageous to use the 7090 version since computation time is substantially less. The 7090 version will be discussed in this appendix.

The program contains two options which are controlled by entering a value of 1 or 0 for KSENS. These options are the computation of $\underline{H}$ or the computation of $\underline{\rho}(k\tau)$. It is necessary to calculate the latter for each control subinterval for use in a later program which calculates the trajectories of the state variables. In case the first column of $\underline{F}$ is zero, which corresponds to an integration in the linear controller, $\underline{\rho}$ will remain constant throughout the control interval; thus its calculation is unnecessary. For this reason the calculation of $\underline{\rho}$ was included in this program only as an option.

The problem, but not the program, is also restartable. It is only necessary to enter the last computed column of $\underline{H}$ as A(K) and start the program. As many more columns of $\underline{H}$ as are desired will then be computed. Finally, when using the Runge-Kutta method one usually starts at $t = 0$; thus the computation of the columns of $\underline{H}$ starts at the right and proceeds to the left.

```
C      CALCULATION OF LARGE A AND H BY RUNGE KUTTA
       READ INPUT TAPE 2,99,N,JE,KA,TAU,DELT,T
       READ INPUT TAPE 2,122,KSENS
   122 FORMAT(I8)
    99 FORMAT(3I4,3F8.0)
       DIMENSION H(6,99),F(6,6),A(6),DX(6,4)
       TAU1=TAU
       DO 1 K=1,N
       DO 1 I=1,N
     1 READ INPUT TAPE 2,112,F(I,K)
   111 FORMAT(E16.8)
   112 FORMAT(F11.0)
       DO 2 K=1,N
     2 READ INPUT TAPE 2,111,A(K)
       P=0.
       IF(KSENS)61,62
    61 PE=0.
    63 FORMAT(21H FOLLOWING IS LARGE A)
       WRITE OUTPUT TAPE 3,63
       GO TO 31
    62 PE=1.
       WRITE OUTPUT TAPE 3,64
    64 FORMAT(25H FOLLOWING IS H TRANSPOSE)
    31 DO 3 K=1,N
       I=K
     3 H(I,JE)=A(K)
     4 DO 6 I=1,N
       Z=0.
       DO 5 K=1,N
     5 Z=Z+F(I,K)*A(K)*DELT
       K=I
     6 DX(K,1)=Z
       DO 8 I=1,N
       Z=0.
       DO 7 K=1,N
     7 Z=Z+F(I,K)*(A(K)+DX(K,1)/2.)*DELT
       K=I
     8 DX(K,2)=Z
       DO 10 I=1,N
       Z=0.
       DO 9 K=1,N
     9 Z=Z+F(I,K)*(A(K)+DX(K,2)/2.)*DELT
       K=I
    10 DX(K,3)=Z
       DO 12 I=1,N
```

```
      Z=0.
      DO 11 K=1,N
   11 Z=Z+F(I,K)*(A(K)+DX(K,3))*DELT
      K=I
   12 DX(K,4)=Z
      DO 13 K=1,N
   13 A(K)=A(K)+((DX(K,1)+2.*DX(K,2)+2.*DX(K,3)+DX(K,4))/6.)
      P=1.+P
  131 IF(TAU-P*DELT) 14,14,4
   14 JE=JE-1
      PE=1.+PE
      TAU=PE*TAU1
      DO 15 K=1,N
      I=K
   15 H(I,JE)=A(K)
      IF(T-TAU) 16,16,4
   16 DO 18 JE=1,KA
   18 WRITE OUTPUT TAPE 3,19,(H(I,JE),I=1,N)
      DO 121 JE=1,KA
      DO 121 I=1,N
  121 WRITE OUTPUT TAPE 14,111,H(I,JE)
   19 FORMAT(6E16.5)
      CALL EXIT
      END
```

The order of the system, the control subinterval, Δt, the number of control subintervals, the control interval, and again the number of control subintervals are read in as N, TAU, DELT, JE, T, and KA respectively. $\underline{F}$ is then read in by statement 1. If $\underline{H}$ is being calculated $\underline{a}$ is read in by statement 2; however, if $\underline{p}$ is being calculated, $\underline{\chi}(o)$ is read in by statement 2. Statements 61 and 62 select the index to be used for the columns of $\underline{H}$ or $\underline{p}$. Statement 3 then begins the Runge-Kutta calculation of $\Delta\underline{\chi}$; $\Delta\underline{x}$ is added to $\underline{\chi}(o)$ by statement 13. Statement 131 then tests to determine whether the value for TAU has been reached. If it has not, control is returned to statement 4 and the calculation of a new value for $\Delta\underline{\chi}$ commences; if TAU has been reached, the $\underline{\chi}$ just calculated is stored as a column of $\underline{H}$ after which the value of TAU is increased. A check is then made to determine if the new value of TAU is the same as T. If it is not control is returned to statement 4; however, if TAU equals T, the matrix $\underline{H}$ is punched on cards by columns one element to each card. $\underline{H}$ is also typed out as $\underline{H}'$. Exactly the same computation is performed for $\underline{p}$ except that different indexing and different initial conditions are used.

Calculation of U

The computation of the optimum control is accomplished by the program which is listed beginning on page 110. The program which is written in 7090 FORTRAN contains two options. These two are the use of the Corner Aiming Descent Scheme or the Gradient Projection Descent Scheme. Refer now to the program listing.

```
C     CALCULATION OF U
      READ INPUT TAPE 2,3,N,KA,MAX,UHI,ULO,EPSIL,DELTA
      READ INPUT TAPE 2,5,KSENS
      DIMENSION XT(6),P(6,6),Y(6),DM(6),A(6)
      DIMENSION U(99),H(6,99),C(99),DELU(99),W(99),ABSC(99)
      DO 2 I=1,N
    2 READ INPUT TAPE 2,7,A(I)
      DO 13 JE=1,KA
   13 READ INPUT TAPE 2,7,U(JE)
    7 FORMAT(E16.8)
    3 FORMAT (I4,I4,I7,F8.1,F8.1,F10.7,F7.2)
      DO 16 JE=1,KA
      DO 16 I=1,N
   16 READ INPUT TAPE 2,7,H(I,JE)
      DO 18 I=1,N
      DO 18 I2=1,N
   18 READ INPUT TAPE 2,4,P(I2,I)
    4 FORMAT(F8.4)
      ITER=0
   20 DO 24 I=1,N
      Z=0.
      DO 23 JE=1,KA
   23 Z=Z+H(I,JE)*U(JE)
   24 XT(I)=Z+A(I)
      V=0.
      DO 35 I2=1,N
      Z=0.
      DO 32 I=1,N
   32 Z=Z+P(I2,I)*XT(I)
      I=I2
      Y(I)=Z
   35 V=V+XT(I)*Z
   34 D=0.
      DO 41 JE=1,KA
      Z=0.
      DO 40 I=1,N
   40 Z=Z+H(I,JE)*Y(I)
   41 C(JE)=Z
      IF(KSENS)199,199,42
   42 DO 46 JE=1,KA
      IF(C(JE)) 43,44,45
   43 W(JE)=UHI-U(JE)
      GO TO 46
   44 W(JE)=0.
      GO TO 46
```

```
 45  W(JE)=ULO-U(JE)
 46  D=D-C(JE)*W(JE)
 47  DO 51  I=1,N
     Z=0.
     DO 50  JE=1,KA
 50  Z=Z+H(I,JE)*W(JE)
 51  DM(I)=Z
     Z3=0.
     DO 58  I2=1,N
     Z2=0.
     DO 55  I=1,N
 55  Z2=Z2+P(I2,I)*DM(I)
     I=I2
 58  Z3=Z3+Z2*DM(I)
     ETA=D/Z3
 63  IF(1.-ETA) 66,64,64
 64  ETA2=ETA
     GO TO 69
 66  ETA2=1.
 69  DO 72  JE=1,KA
 72  U(JE)=U(JE)+ETA2*W(JE)
     GO TO 250
199  Z=0.
200  DO 201  JE=1,KA
201  Z=Z-C(JE)*C(JE)
     ETAN=Z
     DO 203  I=1,N
     Z=0.
     DO 202  JE=1,KA
202  Z=Z+H(I,JE)*C(JE)
203  DM(I)=Z
     ETAD=0.
     DO 205  I2=1,N
     Z=0.
     DO 204  I=1,N
204  Z=Z+P(I2,I)*DM(I)
     I=I2
205  ETAD=ETAD+Z*DM(I)
     ETA=ETAN/ETAD
 53  DO 206  JE=1,KA
     DELU(JE)=ETA*C(JE)
     W(JE)=UHI-U(JE)
     IF(DELU(JE)+U(JE)-UHI) 207,206,206
207  W(JE)=ULO-U(JE)
     IF(DELU(JE)+U(JE)-ULO) 206,206,208
```

```
 208 W(JE)=DELU(JE)
 206 CONTINUE
     ETAN=0.
     DO 209 JE=1,KA
 209 ETAN=ETAN-W(JE)*C(JE)
     DO 211 I=1,N
     Z=0.
     DO 210 JE=1,KA
 210 Z=Z+H(I,JE)*W(JE)
 211 DM(I)=Z
     ETAD=0.
     DO 213 I2=1,N
     Z=0.
     DO 212 I=1,N
 212 Z=Z+P(I2,I)*DM(I)
     I=I2
 213 ETAD=ETAD+Z*DM(I)
     ETA=ETAN/ETAD
     IF(1.-ETA) 214,216,216
 214 ETA=1.
 216 DO 217 JE=1,KA
 217 U(JE)=U(JE)+ETA*W(JE)
 250 ITER=1+ITER
  73 IF(MAX-ITER)971,971,77
  77 DO 97 JE=1,KA
     IF(C(JE)) 81,802,82
  81 ABSC(JE)=-C(JE)
     GO TO 802
  82 ABSC(JE)=C(JE)
 802 IF(U(JE)-UHI) 86,86,84
  84 WRITE OUTPUT TAPE 3,85
  85 FORMAT(20H UH OUTSIDE BOUNDARY)
  86 IF(ULO-U(JE)) 91,91,88
  88 WRITE OUTPUT TAPE 3,89
  89 FORMAT(20H UL OUTSIDE BOUNDARY)
  91 IF(ABSC(JE)-EPSIL) 97,97,93
  93 IF(C(JE)) 94,97,95
  94 IF(UHI-U(JE)-DELTA) 97,97,96
  95 IF(ULO-U(JE)+DELTA) 96,97,97
  96 GO TO 20
  97 CONTINUE
     GO TO 98
 971 WRITE OUTPUT TAPE 3,972
 972 FORMAT(21H ITERATIONS EQUAL MAX)
  98 WRITE OUTPUT TAPE 3,99
```

```
 99 FORMAT(35H ITERATIONS COMPLETE FOLLOWING IS U)
100 DO 101 JE=1,KA
    WRITE OUTPUT TAPE 14,7,U(JE)
101 WRITE OUTPUT TAPE 3,7,U(JE)
    WRITE OUTPUT TAPE 3,5,ITER
  5 FORMAT(I7)
    WRITE OUTPUT TAPE 3,7,V
    CALL EXIT
    END
```

The first READ statement accepts the order of the system, the number of control subintervals, the maximum number of iterations, the upper boundary for control, the lower boundary for control, and the values for ε and δ. These are N, KA, MAX, UHI, ULO, EPSIL, and DELTA respectively. KSENS is read in by the next statement. A KSENS of 0 selects the gradient projection descent scheme while a KSENS of 1 selects the "bang-bang" descent scheme. $\underline{P}$ is next read in by statement 2 as A. Statement 13 accepts U which is the initial guess for the control. The matrices $\underline{H}$ and $\underline{R}$ are read in by statements 16 and 18. The DO loop ending with statement 24 calculates $\underline{\chi}(T)$. The terminal norm V is calculated by the DO loop ending with statement 35. The gradient is calculated next by the DO loop with statement 41. The next statement selects the option of using the Gradient Projection Scheme or the Corner Aiming Scheme. Consider the Corner Aiming Scheme first. The DO loop beginning with statement 42 and ending with 46 examines each component of the gradient vector and determines the proper ω_k according to the Corner Aiming Descent Scheme. Statement 46 then performs the matrix vector multiplication of $\underline{c}'\underline{\omega}$. The DO loops ending with statement 58 then form the product $\underline{\omega}'\underline{H}'\underline{R}\,\underline{H}\underline{\omega}$. The next statement calculates the value η_1, which is always positive. The next four statements determine η_o. Statement 72 then calculates the new value of U. Control is then transferred to statement 250.

If the Gradient Projection Descent Scheme had been selected control would have been transferred from statement 41 to statement 199. The DO loops beginning with 200 and ending with the statement after 205 cal-

culate $\eta = \frac{\underline{g}'\underline{g}}{\underline{g}'H'RH\underline{g}}$. The DO loop beginning with statement 53 then selects the proper values for ω_k in accordance with the Gradient Projection Scheme. A new value for η is then calculated by the next sixteen statements. Statement 214 and the one preceding it compute $sat(\eta)$. The DO loop ending with 217 then computes the new value for U. Statement 250 adds 1 to ITER, which is the number of iterations completed. The value for ITER is then tested to determine if it equals MAX. If it does the value of U just computed is read out. If ITER does not equal MAX the DO loop beginning with statement 77 is entered. This DO loop determines if the conditions of equation (144) for the termination of the descent have been met. If not, control is transferred to statement 20 and another iteration begins. If the conditions of equation (144) have been met, the next few statements read out U, ITER, and V. The values for U are also punched on cards for later use.

Calculation of V

The final program, which is labeled CALCULATION OF V calculates the trajectories of the state variables. This program which is listed on the next two pages is written in 7090 FORTRAN. It requires the $\underline{p}(k\tau)$ matrix, the $\underline{H}$ matrix and the $\underline{R}$ matrix for a particular problem; it then will accept as many different solutions for the control $\underline{u}$ as are desired and from these compute the trajectories for the the state variables for each $\underline{u}$ provided. The trajectories are calculated by solving for the values of

$$\underline{x}(t) = \underline{H}\,\underline{u} + \underline{p}(k\tau) \tag{158}$$

```
C       CALCULATION OF V
        WRITE OUTPUT TAPE 3,20
   20   FORMAT(15H FOLLOWING IS V)
        READ INPUT TAPE 2,99,N,KA,KC
   99   FORMAT(3I6)
        DIMENSION XT(6,80),H(6,83),U(80),P(6,6),A(6,80)
        DO 1 ME=1,KC
        DO 1 I=1,N
    1   READ INPUT TAPE 2,18,A(I,ME)
        DO 2 JE=1,KA
        DO 2 I=1,N
    2   READ INPUT TAPE 2,18,H(I,JE)
        DO 6 I=1,N
        DO 6 I2=1,N
    6   READ INPUT TAPE 2,97,P(I2,I)
   97   FORMAT(F10.0)
   31   READ INPUT TAPE 2,98,NB,KSENS
        K=1
        M=0
        DO 3 JB=1,NB
    3   READ INPUT TAPE 2,18,U(JB)
    9   ME=M+1
        KB=NB-M
        IF(KC-ME)51,52,52
   51   ME=KC
   52   DO 5 I=1,N
        Z=0.
        DO 4 JB=1,KB
        JE=JB+M+KA-NB
    4   Z=Z+H(I,JE)*U(JB)
    5   XT(I,K)=Z+A(I,ME)
        IF(1-K) 71,72,72
   72   V=0.
        DO 7 I2=I,N
        Z=0.
        DO 8 I=1,N
    8   Z=Z+XT(I,1)*P(I2,I)
        I=I2
    7   V=V+Z*XT(I,1)
   71   K=1+K
        M=M+1
   13   IF(NB-M)11,11,9
   11   WRITE OUTPUT TAPE 3,15,V
   41   WRITE OUTPUT TAPE 3,21
   21   FORMAT(30H FOLLOWING IS XT STARTING AT T)
```

```
      DO 16 K=1,NB
   16 WRITE OUTPUT TAPE 3,19,(XT(I,K),I=1,N)
      IF(KSENS)31,31,30
   15 FORMAT(E10.4)
   18 FORMAT(E16.8)
   19 FORMAT(6E12.4)
   98 FORMAT(I8,I4)
   30 CALL EXIT
      END
```

at each control subinterval. In order to compute these trajectories a rather involved system of indexing was necessary.

The first READ statement of the program requires the order of the system, the number of columns of $\underline{H}$ and the number of columns of $\underline{p}(\Delta t)$. In case $\underline{p}(\Delta t)$ is constant throughout the entire control interval it is only necessary to supply one column of $\underline{p}(\Delta t)$; for this reason, the program was designed to use a different number of columns of $\underline{p}(\Delta t)$ than of $\underline{H}$. The READ statements down to statement 6 accept the matrices $\underline{p}(\Delta t)$, $\underline{H}$, and $\underline{R}$. Statement 31 requires a value for the number of cards to be read by statement 3, which accepts the matrix $\underline{u}$. A 1 or a 0 is also required by statement 31 for KSENS. The next ten statements down to statement 5 calculate $\underline{X}(t)$, each element of which is a point in the trajectory of a state variable. The next statement tests to see whether or not $\underline{X}(T)$ for the entire control interval was just computed; if it was, V is calculated by the DO loop ending with statement 7; if $\underline{X}(T)$ was not just calculated, control is transferred to statement 71. In either case, however, new indices are determined by 71 and the following statement. It should be pointed out that the trajectories are calculated beginning at $t = T$ and working back to $t = 0$. This was done to prevent further indexing complications. Statement 13 tests to see if all the points in the trajectory have been computed. If they have not, control is transferred to statement 9 for further computation. If the trajectories are complete they are read out by statement 16. At this point the value for KSENS is tested. If it was zero there will

be another set of trajectories to be calculated for a different control interval; thus, control will be transferred to 31 where new values for the number of control subintervals and for KSENS will be read in. The new control will also be read in by statement 3 and the computations then repeated. If the last value of KSENS was 1, however, exit will be called.

Use of the Programs

The use of the programs is straightforeward and should present no difficulty if the following points are kept in mind:

1) all matrices are read in by columns; each element of a matrix must be on a seperate card;

2) FORMAT for the input statements, if specified, must be followed;

3) all data to be used in another program is punched in the proper FORMAT and in the proper order;

4) DIMENSION statements may have to be changed.

Finally, the programs may be modified for use on other computers. However, since the descent program contains a large number of statements memory limitations of the computer should be kept in mind.

Appendix B

Sample Problem #1

In order to check out the computer programs a sample problem which could be worked out by hand was necessary. It was felt that if an interested reader wished to duplicate the results of this report a statement of this example problem and its solution might be helpful. Following through the solution will also give a better understanding of the calculations of the solutions for a more complex problem.

The problem which was selected is a very simple (from the standpoint of calculation) fourth order problem. The following are the system equations in State Space form:

$$\begin{aligned} \dot{x}_1 &= u \\ \dot{x}_2 &= -\frac{x_2}{\pi} + u \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -x_3 + u \end{aligned} \qquad (159)$$

A control interval of $T = 3\pi$ and a subinterval of $\tau = \frac{\pi}{4}$ were chosen. The $\underline{\Phi}$ and $\underline{H}$ matrices can be calculated by direct integration of the system equations as was developed in Chapter IV. To find the vector $\underline{a}$ the system equations may be integrated with an initial condition of $\underline{x}(0) = \underline{0}$. The system must be excited by a unit step

input. Thus, to find $\underline{a}$ integrate from $t = 0$ to $\frac{\pi}{4}$ the system

$$\begin{aligned}
\dot{x}_1 &= 1 ; & x_1(0) &= 0 \\
\dot{x}_2 &= -\frac{x_2}{\pi} + 1 ; & x_2(0) &= 0 \\
\dot{x}_3 &= x_4 ; & x_3(0) &= 0 \\
\dot{x}_4 &= -x_3 + 1 ; & x_4(0) &= 0
\end{aligned} \tag{160}$$

The integration yields for $\underline{a}$

$$\begin{aligned}
x_1 &= t &&= \pi/4 \\
x_2 &= -\pi(e^{-\frac{1}{4}} - 1) &&= .694 \\
x_3 &= -\cos t + 1 &&= .293 \\
x_4 &= \sin t &&= .707
\end{aligned} \tag{161}$$

Using the above result as an initial condition the following system can be integrated from zero to each control subinterval in order to calculate the matrix $\underline{H}$:

$$\begin{aligned}
\dot{x}_1 &= 0 ; & x_1(0) &= \pi/4 \\
\dot{x}_2 &= -\frac{x_2}{\pi} & x_2(0) &= .694 \\
\dot{x}_3 &= x_4 & x_3(0) &= .293 \\
\dot{x}_4 &= -x_3 & x_4(0) &= .707
\end{aligned} \tag{162}$$

The result of this integration is $\underline{H}'$:

$$\underline{H}' = \begin{bmatrix}
.785 & .095 & .293 & -.707 \\
.785 & .0568 & .707 & -.293 \\
.785 & .0727 & .707 & .293 \\
.785 & .0935 & .293 & .707 \\
.785 & .121 & -.293 & .707 \\
.785 & .155 & -.707 & .293 \\
.785 & .198 & -.707 & -.293 \\
.785 & .255 & -.293 & -.707 \\
.785 & .328 & .293 & -.707 \\
.785 & .420 & .707 & -.293 \\
.785 & .546 & .707 & .293 \\
.785 & .694 & .293 & .707
\end{bmatrix} \tag{163}$$

Since this problem is used to check results of the computation of the optimum driving function, an optimum driving function is chosen using the maximum principal and then the initial error calculated by using this driving function, and setting the solution to the system equations to zero. Figure B-1 shows the driving function chosen.

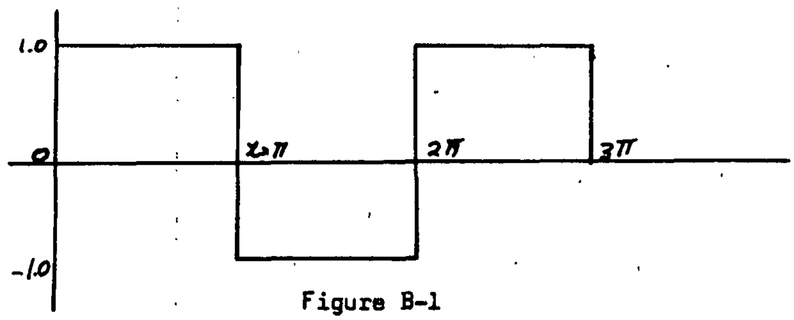

Figure B-1
Optimum Driving Function

This correspond to a $\underline{u}'$ of

$$\underline{u}' = \begin{bmatrix} 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{163}$$

Forming the product $\underline{H}\underline{u}$ gives the vector

$$\underline{H}\underline{u} = \begin{bmatrix} \pi \\ 1.570 \\ 6 \\ 0 \end{bmatrix} \tag{164}$$

The matrix $\underline{\Phi}(3\pi)$ must now be calculated. Again the system equations may be integrated directly by hand to compute $\underline{\Phi}(3\pi)$. The following is the matrix equation which is integrated after first putting it into scalar notation. The matrix $\underline{F}$ is given Table B-I:

$$\frac{d\underline{\Phi}}{dt} = \underline{F}\,\underline{\Phi} \quad ; \quad \underline{\Phi}(0) = \underline{I} \tag{165}$$

The following is obtained for $\underline{\Phi}(3\pi)$

$$\underline{\Phi}(3\pi) = \begin{bmatrix} 1.0 & 0 & 0 & 0 \\ 0 & .05 & 0 & 0 \\ 0 & 0 & -1.0 & 0 \\ 0 & 0 & 0 & -1.0 \end{bmatrix} \tag{166}$$

Now set the solution to the system equal to zero

$$\underline{\Phi}(3\pi)\,\underline{c} + \underline{H}\,\underline{u} = 0 \tag{167}$$

Numerically this is

$$\begin{bmatrix} 1.0 & 0 & 0 & 0 \\ 0 & .05 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1.0 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} \pi \\ 1.570 \\ 6 \\ 0 \end{bmatrix} \tag{168}$$

The initial condition $\underline{c}$ is readily determined to be

$$\underline{c} = \begin{bmatrix} -\pi \\ -30.611 \\ 6 \\ 0 \end{bmatrix} \tag{169}$$

Thus, by working backward from a known solution a problem has been formulated which can be used to test the digital computer programs. The data of Table B-I was used for the computer solutions.

The pertinent results of the computations are shown in Tables B-II and B-III on the last page of this appendix. A slight change had to be made in the control interval and subinterval in order to make subinterval divide into the control interval an integral number

of times. This change accounts for the slight discrepancy in the third decimal place of the $\underline{H}$ matrix. The descent was terminated after 135 iterations when it was evident that the driving function would converge to that calculated above.

Another conclusion may be drawn from the results of this problem in addition to the fact that the computer programs were correct. The same "bang-bang" solution which was used to determine the initial condition was arrived at by an entirely different method when the computer programs were used to find the driving function. The maximum principal was used to formulate the control in the first case. The optimization technique of Steepest Descent was used in the second case. This result again supports the maximum principal of Pontriagin.

Table B-I

Data for Computer Solution

$$\underline{F} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -.318 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$$\underline{g} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \qquad \underline{c} = \begin{bmatrix} -3.1416 \\ -30.611 \\ 6.0 \\ 0.0 \end{bmatrix}$$

T = 9.44
TAV = .786
UHI = 1.0
VLO = -1.0
EPSIL = .002
DELTA = .025

$$\underline{R} = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table B-II

$\underline{H}'$ Computed by Runge-Kutta

```
FØR H SW ØNE ØFF ENTER ALIT AS A
FØR LG A SW ØNE ØN WNTER XNØT AS A
FØLLØWING IS H TRANSPØSE
.786000E+00   .444853E-01   .288202E+00  -.709631E+00
.786000E+00   .571173E-01   .705754E+00  -.297570E+00
.786000E+00   .733363E-01   .709283E+00   .289056E+00
.786000E+00   .941607E-01   .296720E+00   .706112E+00
.786000E+00   .120898E+00  -.289909E+00   .708935E+00
.786000E+00   .155228E+00  -.706468E+00   .295870E+00
.786000E+00   .199306E+00  -.708586E+00  -.290762E+00
.786000E+00   .255901E+00  -.295020E+00  -.706824E+00
.786000E+00   .328567E+00   .291615E+00  -.708235E+00
.786000E+00   .421866E+00   .707178E+00  -.294169E+00
.786000E+00   .541658E+00   .707884E+00   .292467E+00
.786000E+00   .695467E+00   .293318E+00   .707531E+00
STØP
```

Table B-III

Computed Driving Function

```
ITER EQUALS  135
 .99187806E+00
 .99187806E+00
 .99187806E+00
 .99187806E+00
-.98661603E+00
-.99187806E+00
-.99187806E+00
-.96398862E+00
 .99187806E+00
 .99187806E+00
 .99187806E+00
 .99187806E+00
```

Appendix C

Computer Solution of a Fourth Order Test Problem

In order to further test the computer programs and to gain experience in their use a more realistic problem than the one presented in Appendix B was used. The example problem solved by Ho and Brentani in their paper (Ref 5:A-37) was chosen because an answer was readily available. The results are of interest in that a "bang-bang" solution was again obtained.

The data for the programs is set forth in Table C-I. The solution of the optimum driving function for a control interval of 2.5 seconds is presented in Table C-II. These results agree with those obtained by Ho and Brentani.

Table C-I

Data for Computer Programs

$$G_{(s)} = \frac{s + .5}{(s+1)(s+2)(s^2 + 2s + 2)}$$

$$\underline{F} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -4 & -10 & -10 & -5 \end{bmatrix}$$

$$\underline{g} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -4.5 \end{bmatrix} \qquad \underline{c} = \begin{bmatrix} .5 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

T = 25
TAU = .05
UHI = 1.0
ULO = -1.0
EPSIL = .002
DELTA = .025

$$\underline{R} = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table C-II
Optimum Driving Function

ITER EQUALS 50
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.31932941E+00
.69166105E+00
.10000000E+01
.10000000E+01
.10000000E+01
.10000000E+01
.10000000E+01
.10000000E+01
.10000000E+01
.10000000E+01
.35212415E+00
-.10000000E+01
-.10000000E+01
-.10000000E+01
-.10000000E+01

Appendix D

Tabular Data

Tabular data from the computations descrived in Chapter V is presented in this appendix. The first two tables are the matrices for the fourth order approximation and the sixth order system respectively. The rest of the tables are optimum driving functions for various solution and various values of the control interval T. In all the tables of the optimum driving functions the last two numbers from the bottom are the number of iterations used to compute the driving function shown and the value of the terminal norm V, respectively.

Table D-I

$\underline{H}'$ Fourth Order Approximation

FØLLØWING IS H TRANSPØSE

.119739E-01	-.150925E-01	.112162E-01	.318091E-01
.127419E-01	-.156121E-01	.953558E-02	.354125E-01
.135336E-01	-.160431E-01	.767584E-02	.389662E-01
.143445E-01	-.163767E-01	.564029E-02	.424395E-01
.151695E-01	-.166043E-01	.343375E-02	.458005E-01
.160031E-01	-.167174E-01	.106267E-02	.490159E-01
.168392E-01	-.167079E-01	-.146484E-02	.520517E-01
.176717E-01	-.165684E-01	-.413893E-02	.548731E-01
.184938E-01	-.162918E-01	-.694799E-02	.574448E-01
.192985E-01	-.158716E-01	-.987866E-02	.597311E-01
.200784E-01	-.153022E-01	-.129157E-01	.616965E-01
.208261E-01	-.145786E-01	-.160423E-01	.633056E-01
.215337E-01	-.136968E-01	-.192397E-01	.645236E-01
.221931E-01	-.126537E-01	-.224876E-01	.653166E-01
.227963E-01	-.114475E-01	-.257638E-01	.656518E-01
.233351E-01	-.100773E-01	-.290446E-01	.654980E-01
.238013E-01	-.854346E-02	-.323049E-01	.648258E-01
.241867E-01	-.684763E-02	-.355181E-01	.636079E-01
.244834E-01	-.499290E-02	-.386562E-01	.618198E-01
.246834E-01	-.298374E-02	-.416902E-01	.594395E-01
.247793E-01	-.826114E-03	-.445900E-01	.564486E-01
.247637E-01	.147250E-02	-.473246E-01	.528320E-01
.246298E-01	.390307E-02	-.498625E-01	.485786E-01
.243714E-01	.645495E-02	-.521717E-01	.436818E-01
.239825E-01	.911590E-02	-.542200E-01	.381391E-01
.234582E-01	.118720E-01	-.559749E-01	.319533E-01
.227940E-01	.147079E-01	-.574047E-01	.251322E-01
.219864E-01	.176065E-01	-.584778E-01	.176890E-01
.210326E-01	.205492E-01	-.591635E-01	.964257E-02
.199311E-01	.235159E-01	-.594324E-01	.101757E-02
.186810E-01	.264851E-01	-.592562E-01	-.815514E-02
.172829E-01	.294337E-01	-.586084E-01	-.178386E-01
.157384E-01	.323376E-01	-.574645E-01	-.279896E-01
.140503E-01	.351715E-01	-.558024E-01	-.385586E-01
.122228E-01	.379089E-01	-.536026E-01	-.494896E-01
.102615E-01	.405225E-01	-.508465E-01	-.607204E-01
.817317E-02	.429843E-01	-.475267E-01	-.721826E-01
.596611E-02	.452655E-01	-.436276E-01	-.838016E-01
.365010E-02	.473373E-01	-.391453E-01	-.954970E-01
.123636E-02	.491703E-01	-.340781E-01	-.107182E+00

STØP

Table D-II

$\underline{H}'$, Sixth Order System

-0.13026E-02	0.56834E-03	0.20475E-03	-0.27215E-03	0.22965E-03	-0.23297E-03
0.24350E-03	-0.28165E-03	0.17471E-03	0.63322E-03	-0.12910E-02	-0.33886E-03
0.25060E-03	-0.28582E-03	0.15848E-03	0.66538E-03	-0.12811E-02	-0.44735E-03
0.25779E-03	-0.28957E-03	0.14144E-03	0.69726E-03	-0.12686E-02	-0.55833E-03
0.26507E-03	-0.29288E-03	0.12362E-03	0.72878E-03	-0.12532E-02	-0.67165E-03
0.27317E-03	-0.29600E-03	0.10311E-03	0.76298E-03	-0.12330E-02	-0.79885E-03
0.27986E-03	-0.29813E-03	0.85628E-04	0.79051E-03	-0.12138E-02	-0.90476E-03
0.28733E-03	-0.30002E-03	0.65488E-04	0.82056E-03	-0.11897E-02	-0.10242E-02
0.29485E-03	-0.30140E-03	0.44605E-04	0.84997E-03	-0.11626E-02	-0.11454E-02
0.30240E-03	-0.30224E-03	0.22996E-04	0.87867E-03	-0.11324E-02	-0.12682E-02
0.31072E-03	-0.30254E-03	-0.15912E-05	0.90931E-03	-0.10957E-02	-0.14047E-02
0.31752E-03	-0.30227E-03	-0.22325E-04	0.93360E-03	-0.10628E-02	-0.15175E-02
0.32507E-03	-0.30142E-03	-0.45993E-04	0.95968E-03	-0.10233E-02	-0.16437E-02
0.33259E-03	-0.29997E-03	-0.70301E-04	0.98474E-03	-0.98063E-03	-0.17707E-02
0.34006E-03	-0.29790E-03	-0.95221E-04	0.10087E-02	-0.93477E-03	-0.18982E-02
0.34748E-03	-0.29520E-03	-0.12073E-03	0.10314E-02	-0.88572E-03	-0.20260E-02
0.35482E-03	-0.29186E-03	-0.14678E-03	0.10529E-02	-0.83347E-03	-0.21538E-02
0.36278E-03	-0.28742E-03	-0.17605E-03	0.10750E-02	-0.77231E-03	-0.22942E-02
0.36920E-03	-0.28318E-03	-0.20043E-03	0.10918E-02	-0.71940E-03	-0.24088E-02
0.37622E-03	-0.27783E-03	-0.22794E-03	0.11090E-02	-0.65759E-03	-0.25353E-02
0.38309E-03	-0.27178E-03	-0.25586E-03	0.11247E-02	-0.59264E-03	-0.26610E-02
0.38980E-03	-0.26503E-03	-0.28416E-03	0.11386E-02	-0.52455E-03	-0.27854E-02
0.39698E-03	-0.25679E-03	-0.31566E-03	0.11520E-02	-0.44609E-03	-0.29206E-02
0.40268E-03	-0.24939E-03	-0.34169E-03	0.11613E-02	-0.37915E-03	-0.30296E-02
0.40940E-03	-0.23956E-03	-0.37376E-03	0.11706E-02	-0.29403E-03	-0.31605E-02
0.41469E-03	-0.23085E-03	-0.40016E-03	0.11764E-02	-0.22173E-03	-0.32655E-02
0.42034E-03	-0.22048E-03	-0.42963E-03	0.11809E-02	-0.13866E-03	-0.33797E-02
0.42623E-03	-0.20822E-03	-0.46215E-03	0.11834E-02	-0.44030E-04	-0.35019E-02
0.43080E-03	-0.19752E-03	-0.48878E-03	0.11835E-02	0.35861E-04	-0.35990E-02
0.43604E-03	-0.18363E-03	-0.52130E-03	0.11811E-02	0.13642E-03	-0.37137E-02
0.44004E-03	-0.17160E-03	-0.54783E-03	0.11771E-02	0.22100E-03	-0.38041E-02
0.44415E-03	-0.15754E-03	-0.57718E-03	0.11704E-02	0.31732E-03	-0.39006E-02
0.44826E-03	-0.14123E-03	-0.60923E-03	0.11602E-02	0.42599E-03	-0.40015E-02
0.45129E-03	-0.12722E-03	-0.63522E-03	0.11496E-02	0.51691E-03	-0.40798E-02
0.45454E-03	-0.10932E-03	-0.66662E-03	0.11338E-02	0.63036E-03	-0.41699E-02
0.45683E-03	-0.94038E-04	-0.69197E-03	0.11186E-02	0.72496E-03	-0.42387E-02
0.45915E-03	-0.74589E-04	-0.72244E-03	0.10970E-02	0.84261E-03	-0.43165E-02
0.46079E-03	-0.56187E-04	-0.74959E-03	0.10746E-02	0.95134E-03	-0.43808E-02
0.46186E-03	-0.39052E-04	-0.77352E-03	0.10521E-02	0.10505E-02	-0.44333E-02
0.46259E-03	-0.19388E-04	-0.79948E-03	0.10244E-02	0.11620E-02	-0.44853E-02
0.46282E-03	0.91634E-06	-0.82471E-03	0.99395E-03	0.12747E-02	-0.45304E-02
0.46254E-03	0.21841E-04	-0.84915E-03	0.96067E-03	0.13885E-02	-0.45682E-02
0.46161E-03	0.45551E-04	-0.87503E-03	0.92075E-03	0.15146E-02	-0.46010E-02
0.46037E-03	0.65470E-04	-0.89536E-03	0.88551E-03	0.16183E-02	-0.46208E-02
0.45845E-03	0.88126E-04	-0.91698E-03	0.84360E-03	0.17340E-02	-0.46352E-02
0.45596E-03	0.11131E-03	-0.93751E-03	0.79880E-03	0.18500E-02	-0.46413E-02
0.45288E-03	0.13499E-03	-0.95689E-03	0.75110E-03	0.19660E-02	-0.46389E-02
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

[illegible]	[illegible]	[illegible]	0.11802E-02	0.42599E-03	-0.40015E-02
0.45129E-03	-0.12722E-03	-0.63522E-03	0.11496E-02	0.51691E-03	-0.40798E-02
0.45454E-03	-0.10932E-03	-0.66662E-03	0.11338E-02	0.63036E-03	-0.41699E-02
0.45683E-03	-0.94038E-04	-0.69197E-03	0.11186E-02	0.72496E-03	-0.42387E-02
0.45915E-03	-0.74589E-04	-0.72244E-03	0.10970E-02	0.84261E-03	-0.43165E-02
0.46079E-03	-0.56187E-04	-0.74959E-03	0.10746E-02	0.95134E-03	-0.43808E-02
0.46186E-03	-0.39052E-04	-0.77352E-03	0.10521E-02	0.10505E-02	-0.44333E-02
0.46259E-03	-0.19388E-04	-0.79948E-03	0.10244E-02	0.11620E-02	-0.44853E-02
0.46282E-03	0.91634E-06	-0.82471E-03	0.99395E-03	0.12747E-02	-0.45304E-02
0.46254E-03	0.21841E-04	-0.84915E-03	0.96067E-03	0.13885E-02	-0.45682E-02
0.46161E-03	0.45551E-04	-0.87503E-03	0.92075E-03	0.15146E-02	-0.46010E-02
0.46037E-03	0.65470E-04	-0.89536E-03	0.88551E-03	0.16183E-02	-0.46208E-02
0.45845E-03	0.88126E-04	-0.91698E-03	0.84360E-03	0.17340E-02	-0.46352E-02
0.45596E-03	0.11131E-03	-0.93751E-03	0.79880E-03	0.18500E-02	-0.46413E-02
0.45288E-03	0.13499E-03	-0.95689E-03	0.75110E-03	0.19660E-02	-0.46389E-02
0.44920E-03	0.15914E-03	-0.97504E-03	0.70050E-03	0.20819E-02	-0.46276E-02
0.44445E-03	0.18622E-03	-0.99350E-03	0.64151E-03	0.22088E-02	-0.46039E-02
0.44001E-03	0.20873E-03	-0.10074E-02	0.59064E-03	0.23121E-02	-0.45745E-02
0.43448E-03	0.23409E-03	-0.10214E-02	0.53142E-03	0.24259E-02	-0.45295E-02
0.42831E-03	0.25978E-03	-0.10339E-02	0.46936E-03	0.25385E-02	-0.44713E-02
0.42149E-03	0.28577E-03	-0.10448E-02	0.40450E-03	0.26494E-02	-0.44043E-02
0.41323E-03	0.31465E-03	-0.10550E-02	0.32999E-03	0.27696E-02	-0.43390E-02
0.40503E-03	0.34112E-03	-0.10623E-02	0.25940E-03	0.28777E-02	-0.43187E-02
0.39709E-03	0.36508E-03	-0.10674E-02	0.19355E-03	0.29752E-02	-0.43583E-02
0.38763E-03	0.39182E-03	-0.10713E-02	0.11780E-03	0.30853E-02	-0.44533E-02
0.37750E-03	0.41263E-03	-0.10733E-02	0.39270E-04	0.31972E-02	-0.44552E-02
0.36558E-03	0.44815E-03	-0.10731E-02	-0.50292E-04	0.33138E-02	-0.38536E-02
0.35523E-03	0.47228E-03	-0.10712E-02	-0.12572E-03	0.33850E-02	-0.22630E-02
0.34308E-03	0.49901E-03	-0.10670E-02	-0.21074E-03	0.34018E-02	0.12295E-02
0.33028E-03	0.52561E-03	-0.10606E-02	-0.29492E-03	0.33135E-02	0.58120E-02
0.31680E-03	0.55202E-03	-0.10523E-02	-0.37563E-03	0.31367E-02	0.71145E-02
0.30268E-03	0.57820E-03	-0.10419E-02	-0.45259E-03	0.30719E-02	-0.55055E-02
0.28790E-03	0.60410E-03	-0.10296E-02	-0.53458E-03	0.36642E-02	-0.48271E-01
0.27247E-03	0.62966E-03	-0.10149E-02	-0.64892E-03	0.58308E-02	-0.13130E-00
0.25478E-03	0.65731E-03	-0.99430E-03	-0.87276E-03	0.10923E-01	-0.23219E-00
0.23974E-03	0.67943E-03	-0.97145E-03	-0.11790E-02	0.16218E-01	-0.20698E-00
0.22245E-03	0.70331E-03	-0.93652E-03	-0.16215E-02	0.17528E-01	0.19670E-00
0.20458E-03	0.72617E-03	-0.89159E-03	-0.19015E-02	0.69230E-04	0.13518E 01
0.18615E-03	0.74790E-03	-0.84873E-03	-0.12888E-02	-0.57560E-01	0.33719E 01
0.16719E-03	0.76892E-03	-0.84472E-03	0.14308E-02	-0.16809E-00	0.52570E 01
0.14770E-03	0.79101E-03	-0.94677E-03	0.72588E-02	-0.29156E-00	0.35583E 01
0.12760E-03	0.81774E-03	-0.12235E-02	0.14776E-01	-0.26018E-00	-0.85067E 01
0.10674E-03	0.85341E-03	-0.16364E-02	0.16007E-01	0.28604E-00	-0.38842E 02
0.84856E-04	0.89780E-03	-0.18166E-02	-0.80186E-02	0.18346E 01	-0.86514E 02
0.61902E-04	0.93437E-03	-0.78768E-03	-0.85524E-01	0.45105E 01	-0.11970E 03
0.38602E-04	0.91370E-03	0.30563E-02	-0.23246E-00	0.69463E 01	-0.45358E 02
0.17434E-04	0.74657E-03	0.11001E-01	-0.39358E-00	0.45104E 01	0.30174E 03
0.24798E-05	0.29010E-03	0.21865E-01	-0.30997E-00	-0.14639E 02	0.11829E 04

Table D-III

Optimum Control for Sixth Order System

Direct Control of First Two State Variables

T = .675 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.10524182E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
   129
 0.12030042E-00
```

Table D-III
(cont.)

Optimum Control for Sixth Order System

Direct Control of First Two State Variables

T = .625 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.12865712E 02
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
   60
 0.35551506E-00
```

Table D-III
(cont.)

Optimum Control for Sixth Order System

Direct Control of First Two State Variables

T = .575 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
 0.95030949E 02
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
    65
 0.69346071E 00
```

Table D-III
(cont.)

Optimum Control for Sixth Order System

Direct Control of First Two State Variables

T = .500 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
 0.54806840E 02
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
     44
 0.13508035E 01
```

Table D-III
(cont.)

Optimum Control for Sixth Order System

Direct Control of First Two State Variables

T = .375 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
-0.19500000E 03
 0.18410978E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
 0.19500000E 03
    33
 0.26502431E 01
```

Table D- IV

Optimum Control for Fourth Order Approximation

Direct Control of First Two State Variables

T = .650 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
 0.23082887E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
    62
 0.16079205E-00
```

T = .600 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.35306581E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
    21
 0.41684502E-00
```

Table D-IV

Optimum Control for Fourth Order Approximation

Direct Control of First Two State Variables

T = .500 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
 0.82834515E 00
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
    32
 0.12158052E 01
```

Table D-V

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = 1.85 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.65408815E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
-0.34156581E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
  1350
 0.81695723E-01
```

Table D-V
(cont.)

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = 1.75 sec.

```
ITERATIONS COMPLETE FOLLOWING,IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.18881381E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.34043817E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
   695
 0.24872325E-00
```

Table D-V
(cont.)

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = 1.50 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.22782274E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.67485256E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
   355
 0.90023644E 00
```

Table D-V
(cont.)

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = 1.25 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
 0.55472609E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
-0.39656283E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
   290
 0.16630284E 01
```

Table D-V
(cont.)

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = 1.00 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.68830354E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.16569404E-00
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
   460
 0.23394992E 01
```

Table D-V
(cont.)

Optimum Control for Fourth Order Approximation

Direct Control of All Four State Variables

T = .75 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
 0.36115248E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.53904514E 01
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
   337
 0.28406490E 01
```

T = .50 sec.

```
ITERATIONS COMPLETE FOLLOWING IS U
-0.70999999E 01
-0.70999999E 01
-0.70999999E 01
-0.52056129E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
 0.70999999E 01
-0.19497150E 01
-0.70999999E 01
   754
 0.31596032E 01
```

Table D-VI

State Variable Trajectories

Sixth Order System

Direct Control of Two State Variables

FOLLOWING IS XT STARTING AT T

0.1337E-00	-0.1657E-00	0.9034E 01	-0.1988E 03	0.4885E 03	0.2435E 06
0.1406E-00	-0.3948E-00	0.9162E 01	-0.1982E 03	0.5086E 03	0.2423E 06
0.1533E-00	-0.6269E 00	0.9285E 01	-0.1986E 03	0.5686E 03	0.2404E 06
0.1718E-00	-0.8625E 00	0.9442E 01	-0.2010E 03	0.6748E 03	0.2390E 06
0.1963E-00	-0.1103E 01	0.9695E 01	-0.2063E 03	0.7747E 03	0.2418E 06
0.2270E-00	-0.1352E 01	0.1010E 02	-0.2123E 03	0.6862E 03	0.2554E 06
0.2639E-00	-0.1612E 01	0.1058E 02	-0.2101E 03	0.4974E 02	0.2863E 06
0.3075E-00	-0.1881E 01	0.1071E 02	-0.1806E 03	-0.1556E 04	0.3303E 06
0.3578E-00	-0.2140E 01	0.9499E 01	-0.9816E 02	-0.4090E 04	0.3501E 06
0.4139E-00	-0.2336E 01	0.5433E 01	0.4459E 02	-0.5879E 04	0.2486E 06
0.4732E-00	-0.2380E 01	-0.2395E 01	0.1737E 03	-0.1830E 04	-0.1377E 06
0.5318E 00	-0.2293E 01	-0.4561E 01	0.2014E 03	-0.5156E 03	-0.2438E 06
0.5877E 00	-0.2175E 01	-0.4761E 01	0.2013E 03	-0.5149E 03	-0.2438E 06
0.6406E 00	-0.2051E 01	-0.4959E 01	0.2012E 03	-0.5137E 03	-0.2439E 06
0.6903E 00	-0.1923E 01	-0.5153E 01	0.2010E 03	-0.5116E 03	-0.2439E 06
0.7368E 00	-0.1790E 01	-0.5342E 01	0.2007E 03	-0.5083E 03	-0.2440E 06
0.7799E 00	-0.1653E 01	-0.5524E 01	0.2004E 03	-0.5050E 03	-0.2439E 06
0.8195E 00	-0.1511E 01	-0.5697E 01	0.2001E 03	-0.5056E 03	-0.2436E 06
0.8555E 00	-0.1365E 01	-0.5862E 01	0.1998E 03	-0.5181E 03	-0.2429E 06
0.8878E 00	-0.1214E 01	-0.6027E 01	0.2002E 03	-0.5530E 03	-0.2419E 06
0.9163E 00	-0.1060E 01	-0.6215E 01	0.2017E 03	-0.6111E 03	-0.2413E 06
0.9408E 00	-0.8999E 00	-0.6459E 01	0.2047E 03	-0.6582E 03	-0.2433E 06
0.9613E 00	-0.7328E 00	-0.6785E 01	0.2077E 03	-0.5863E 03	-0.2515E 06
0.9775E 00	-0.5571E 00	-0.7135E 01	0.2054E 03	-0.1940E 03	-0.2691E 06
0.9892E 00	-0.3746E-00	-0.7253E 01	0.1869E 03	0.7322E 03	-0.2924E 06
0.9964E 00	-0.1977E-00	-0.6564E 01	0.1388E 03	0.2102E 04	-0.2983E 06
0.9995E 00	-0.5657E-01	-0.4264E 01	0.6044E 02	0.2855E 04	-0.2307E 06

Table D-VII

State Variable Trajectories

Fourth Order Approximation

Direct Control of Two State Variables

FOLLOWING IS XT STARTING AT T

0.1582E-00	-0.1780E-00	0.1994E 01	-0.1860E 01
0.1786E-00	-0.6403E 00	0.2285E 01	-0.8368E 00
0.2224E-00	-0.1116E 01	0.2521E 01	0.3123E-00
0.2903E-00	-0.1601E 01	0.2698E 01	0.1566E 01
0.3827E-00	-0.2094E 01	0.2808E 01	0.2920E 01
0.4938E-00	-0.2348E 01	0.2713E 01	0.3802E 01
0.6055E 00	-0.2118E 01	0.2305E 01	0.3603E 01
0.7053E 00	-0.1869E 01	0.1909E 01	0.3329E 01
0.7921E 00	-0.1599E 01	0.1528E 01	0.2978E 01
0.8649E 00	-0.1312E 01	0.1167E 01	0.2547E 01
0.9229E 00	-0.1007E 01	0.8296E 00	0.2034E 01
0.9653E 00	-0.6852E 00	0.5199E 00	0.1439E 01
0.9912E 00	-0.3491E-00	0.2420E-00	0.7610E 00

Table D-VIII

State Variable Trajectories

Fourth Order Approximation

Direct Control of Four State Variables

FOLLOWING IS XT STARTING AT T

0.1221E-00	-0.7038E-03	0.8481E-01	-0.8755E-01
0.1133E-00	0.3552E-00	-0.1129E-00	-0.9223E 00
0.8645E-01	0.7199E 00	-0.2668E-00	-0.1837E 01
0.4120E-01	0.1091E 01	-0.3731E-00	-0.2829E 01
-0.2272E-01	0.1466E 01	-0.4278E-00	-0.3896E 01
-0.1055E-00	0.1843E 01	-0.4275E-00	-0.5033E 01
-0.2070E-00	0.2218E 01	-0.3686E-00	-0.6238E 01
-0.3272E-00	0.2589E 01	-0.2478E-00	-0.7504E 01
-0.4657E-00	0.2952E 01	-0.6238E-01	-0.8826E 01
-0.6175E 00	0.3117E 01	0.2951E-00	-0.9759E 01
-0.7638E 00	0.2730E 01	0.9983E 00	-0.9442E 01
-0.8899E 00	0.2307E 01	0.1682E 01	-0.8998E 01
-0.9940E 00	0.1852E 01	0.2341E 01	-0.8421E 01
-0.1075E 01	0.1364E 01	0.2967E 01	-0.7708E 01
-0.1130E 01	0.8456E 00	0.3555E 01	-0.6856E 01
-0.1159E 01	0.2992E-00	0.4096E 01	-0.5863E 01
-0.1159E 01	-0.2731E-00	0.4584E 01	-0.4727E 01
-0.1131E 01	-0.8682E 00	0.5011E 01	-0.3449E 01
-0.1072E 01	-0.1483E 01	0.5371E 01	-0.2030E 01
-0.9824E 00	-0.2114E 01	0.5657E 01	-0.4713E-00
-0.8606E 00	-0.2757E 01	0.5862E 01	0.1223E 01
-0.7065E 00	-0.3409E 01	0.5978E 01	0.3049E 01
-0.5369E 00	-0.3372E 01	0.5612E 01	0.3379E 01
-0.3704E-00	-0.3287E 01	0.5215E 01	0.3606E 01
-0.2085E-00	-0.3183E 01	0.4807E 01	0.3785E 01
-0.5241E-01	-0.3058E 01	0.4392E 01	0.3910E 01
0.9692E-01	-0.2912E 01	0.3972E 01	0.3979E 01
0.2384E-00	-0.2745E 01	0.3550E 01	0.3986E 01
0.3711E-00	-0.2557E 01	0.3129E 01	0.3928E 01
0.4938E-00	-0.2348E 01	0.2713E 01	0.3802E 01
0.6055E 00	-0.2118E 01	0.2305E 01	0.3603E 01
0.7053E 00	-0.1869E 01	0.1909E 01	0.3329E 01
0.7921E 00	-0.1599E 01	0.1528E 01	0.2978E 01
0.8649E 00	-0.1312E 01	0.1167E 01	0.2547E 01
0.9229E 00	-0.1007E 01	0.8296E 00	0.2034E 01
0.9653E 00	-0.6852E 00	0.5199E 00	0.1439E 01
0.9912E 00	-0.3491E-00	0.2420E-00	0.7610E 00